PC 98
System Design Guide

A Technical Reference for Designing PCs and Peripherals for the Microsoft Windows Family of Operating Systems

Intel Corporation and Microsoft Corporation
With special contributions by Compaq Computer Corporation

Contents

Welcome

This guide is for engineers who build personal computers, expansion cards, and peripheral devices that will be used with the Microsoft® Windows® 98 and Windows NT® version 5.0 operating systems. The goal of this document is to provide hardware design guidelines that will result in the optimal user experience, particularly when the hardware is used with the Windows family of operating systems.

This guide is co-authored by Intel Corporation and Microsoft Corporation, with contributions by Compaq Computer Corporation. The requirements and recommendations in this guide outline features that the hardware industry should consider in designing PCs and peripherals for various price levels and performance levels.

The clarifications, changes, and additional requirements in this guide define the PC 98 requirements for 1998–99 in the way that *PC 97 Hardware Design Guide* (Microsoft Press, 1996; ISDN 1-57231-381-1) defined the PC 97 requirements for 1997–98.

This guide includes PC 98 requirements for basic consumer and office implementations, such as desktop, mobile, and workstation systems, and for Entertainment PCs. In this guide, the following requirements are defined:

- Design requirements for specific types of systems that will run either Windows 98 or Windows NT operating systems
- Design requirements related to the OnNow design initiative, including requirements related to the Advanced Configuration and Power Interface (ACPI) specification, Plug and Play device configuration, and power management in PC systems
- New manageability requirements that focus on improving Windows 98 and Windows NT, with the end goal of reducing total cost of ownership (TCO) by providing support for maximum automation of administrative tasks using centralized control and maximum flexibility
- Clarifications and additional design requirements for devices supported under Windows 98 and Windows NT, including new graphics and video device capabilities, DVD, scanners and digital cameras, and other devices

This book does not address PC systems designed to act as servers in networked environments. It also does not address non-PC handheld computers running on the Microsoft Windows CE operating system.

Important: The system requirements defined in this document provide guidelines for designing PC systems that deliver an enhanced user experience when implemented with Windows 98 and Windows NT operating systems. These design requirements are not related to the minimum, most-optimal, or best system requirements for running the Windows family of operating systems. For information about minimum system requirements, see the web site at http://www.microsoft.com/windows/.

How to Use This Guide

The PC 98 requirements are defined by system type and for individual bus classes and device classes. This guide is divided into five parts, with each part addressing a particular element of PC 98 design.

Part 1: System Design Issues. Introduces the important design issues for PC 98. Study this part first to understand the key design issues being addressed in the PC 98 requirements.

Part 2: PC 98 Systems. Presents system-type definitions and PC 98 requirements for each system type. Study this part for an understanding of the overall system requirements.

Part 3: PC 98 Bus Design Guidelines. Presents requirements for each bus type and I/O host controller supported under Windows 98 and Windows NT. Study this part for a detailed understanding of how buses and controllers are to be implemented on PC 98 systems.

Part 4: Device Design Guidelines. Defines design requirements for each particular device type, whether the device is an integral part of a PC system or designed as an add-on device. Study this part for a detailed understanding of the design requirements for each device type.

References. Includes the PC 98 checklist, which summarizes all the requirements defined in this guide, plus other technical appendixes, a comprehensive hardware glossary with a list of acronyms and abbreviations used in this guide, and an index.

Updates to this guide, technical clarifications, and answers to frequently asked questions are available on the following web sites:

http://www.microsoft.com/hwdev/desguid/
http://developer.intel.com/design/pc98/

Required vs. Recommended PC 98 Features

In this guide, hardware features are described as *Required*, *Recommended*, or *Optional.* For PC 98, these terms are used to mean the following:

- **Required:** These basic features must be implemented in order for hardware to comply with PC 98 requirements.
- **Recommended:** These features add capabilities that are supported by the Windows family of operating systems. Recommended features take advantage of the native capabilities of the device drivers included with the operating system, usually without imposing major cost increases.

  Notice that for compliance testing, if a recommended feature is implemented, it must meet the requirements for that feature as defined in this guide.

  Some recommended features might become requirements in the future.
- **Optional:** These features are neither required nor recommended, but if the feature is implemented in a PC 98 system, it must meet the specified requirements. Optional features will not become requirements in the future.

In this guide, these words can be understood as follows with regard to PC 98 requirements:

- **Must:** Required
- **Should:** Recommended

Important: The requirements and recommendations in this guide are often provided in the form of references to industry specifications. These specifications might contain intellectual property of Intel, Microsoft, or other third parties. Each of these industry specifications might have different intellectual property licensing arrangements. It is the original equipment manufacturer’s (OEM’s) responsibility to consult these industry specifications or their issuance bodies for licensing specifics or details.

Conventions Used in This Guide

The following conventional terms are used throughout this guide. In addition, see the Hardware Glossary in the References part of this guide.

Convention	Meaning
Add-on device	Refers to devices that are traditionally added to the basic PC system to increase functionality. Examples include audio, networking, graphics, small computer system interface (SCSI) controller, and so on. Add-on devices fall into two categories: devices built on to the system board and devices on expansion cards added to the system through a system-board connector, such as Peripheral Component Interconnect (PCI).
Intel Architecture	Refers to computers based on 32-bit microprocessors that use the Intel Architecture instruction set, such as Intel® 80486, Intel Pentium®, Intel Pentium with MMX™ technology, Pentium Pro, Pentium II, or compatible processors. MMX technology refers to Intel’s media-enhancement technology that includes new instructions to the Intel Architecture instruction set.
PC 97	Collection of requirements and recommendations for PC system design defined in the *PC 97 Hardware Design Guide*.
PC 98	Collection of requirements and recommendations defined in this guide that make up the 1998–99 requirements for PC system design. **Note:** The term “PC 98” as used in this guide should not be confused with NEC PC98, a Japanese-specific product.
RISC-based	Refers to computers based on reduced instruction set computing (RISC) architecture. Notice that all requirements and recommendations for RISC-based PCs are for the Windows NT operating system only.
System device	Also *on-board device.* Refers to devices on the system board such as interrupt controllers, keyboard controller, real-time clock, direct memory access (DMA) page registers, DMA controllers, memory controllers, floppy disk controller (FDC), Integrated Device Electronics (IDE) ports, serial and parallel ports, PCI bridges, and so on. In today’s PCs, these devices are typically integrated with the supporting chip set.
Windows	For PC 98, refers to the Microsoft Windows 98 operating system, including any add-on capabilities and any later versions of the operating system.
Windows NT	For PC 98, refers to the Microsoft Windows NT Workstation version 5.0 operating system, including any add-on capabilities and any later versions of the operating system.

PC 98 and the “Designed for Microsoft Windows” Logo Program

Microsoft will use the requirements and recommendations in this guide as the basis of requirements for the 1998–99 “Designed for Microsoft Windows” hardware logo program.

The “Designed for Microsoft Windows” logo program—represented in 1997 by the “Designed for Microsoft Windows NT and Windows 95” logo program—was developed by Microsoft to help end users easily identify hardware and software products designed specifically for the Windows and Windows NT operating systems. Users can mix and match hardware and software products designated with these logos and can be assured that the products will take advantage of the new technologies integrated into the Microsoft Windows family of 32-bit operating systems. The goal is to ensure that end users will have an optimal experience using the software and hardware products that carry this logo.

The logo program provides customers with the assurance that their hardware works with the Windows and Windows NT family of products, with an emphasis on how the system performs when running commercially marketed desktop applications. The end result Microsoft is seeking is lower cost of support for both vendors and users.

Licensing the logo enables vendors to use the logo on web sites, product packaging, advertising, collateral, and other marketing materials. The logo indicates to customers that the product is designed to meet a specific set of standards and to provide an optimal experience when run on either a Windows or Windows NT operating system.

Logo Compliance Dates. In general, the PC 98 requirements go into effect on July 1, 1998, for the “Designed for Microsoft Windows” logo. Compliance testing for some requirements will begin later because of the time required for silicon changes to become widely available. For information about actual compliance testing dates for specific requirements, see the web site at http://www.microsoft.com/hwdev/desguid/.

Logo Testing. Both hardware and software are tested before rights to use the "Designed for Microsoft Windows" logo are granted. The testing organization for the logo program is the Windows Hardware Quality Labs (WHQL), which provides compatibility testing services for Windows and Windows NT hardware and drivers.

Hardware developers whose products pass the WHQL testing program also receive a detailed test report, inclusion of tested hardware on the Windows Hardware Compatibility List (HCL), and free distribution of drivers in the Windows Driver Library (WDL).

If you have questions about the program, contact WHQL:

Windows Hardware Quality Labs
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399 USA

http://www.microsoft.com/hwtest/
E-mail: whqlinfo@microsoft.com
Fax: (425) 703-3872

References

The following table lists some of the information resources, services, and tools available from Intel and Microsoft to help build hardware that is compliant with the PC 98 requirements. In addition, each chapter in this guide contains a reference section.

Resource	Address
Intel information for developers	http://developer.intel.com
Microsoft information for hardware manufacturers	http://www.microsoft.com/hwdev/ E-mail: ihv@microsoft.com
Windows and Windows NT DDKs	Provided with Microsoft Developer Network (MSDN) Professional membership. To subscribe: Fax: (425) 936-7329, Attn: Developer Network E-mail: msdn@microsoft.com http://www.microsoft.com/msdn/subscribe/
Plug and Play and Microsoft-provided specifications	http://www.microsoft.com/hwdev/specs/
Hardware testing tools	http://www.microsoft.com/hwtest/

Acknowledgments

The editors wish to thank the numerous individuals within Intel and Microsoft who contributed material and review comments to this design guide.

Intel and Microsoft also gratefully acknowledge the input received from the representatives of the many companies who reviewed early drafts of this guide, including the following:

3Com Corporation
3D Labs
Accelerix, Inc.
Acer, Inc.
Adaptec, Inc.
Advanced Micro Devices
Alpine Software Technologies
Association for Computing Machinery
AST Research
ATI Instruments
AVM Berlin
Azfin Semiconductors Pte, Limited
Aztech Systems, Limited
Brooktree Corporation
Canon, Inc.
Chips and Technologies, Inc.
Cirrus Logic, Inc.
Compaq Computer Corporation
Creative Labs
Chromatic Research, Inc.
Crystal Semiconductor
Cyrix Corporation
Dell Computer Corporation
Digital Equipment Corporation
Digital Infotainment
Efficient Networks, Inc.
ELSA, Inc.
Epson
ESS Technology, Inc.
Evans & Sutherland Computer Corporation
Fujitsu, Limited
Gamry Instruments, Inc.
Gateway 2000, Inc.

Hewlett-Packard Company
Hirin Information Technology Co., Limited
Hitachi, Limited
Integrated Device Technology, Inc.
International Business Machines Corporation
Information Technology and Telecommunications Equipment
(ITTE) Intercommittee Working Group on Acoustics,
including:

- Information Technology Industries Council, TC-6, Product Acoustics
- Institute of Noise Control Engineering, Technical Committee on Information Technology and Telecommunications Equipment
- ISO Technical Committee 43/SC1/WG23 Noise from Information Technology and Telecommunications Equipment
- ANSI S12-3, Noise from ITTE
- ECMA, TC 26, Noise

Iomega Corporation
Iterated Systems
ITT Semiconductors
Kasan Electronics
Keesler AFB
Madge Networks
Matrox Graphics, Inc.
Matsushita Electric Industrial Co., Limited
Melco, Inc.
Micron Electronics, Inc.
Mitsubishi Electric Corporation
NEC Corporation
NeoMagic Corporation
Nihon Unisys, Limited
Nissei Electric Company, Limited
Norpak Corporation
NVidia
Oak Leaf Systems, Inc.
Packard Bell NEC, Inc.
Phoenix Technologies
Pioneer Electronic Corporation
Quantum Corporation
Real3D/Realsoft International
Rockwell Semiconductor Systems
Roger Jennings, consultant

SBC Communications
Schlumberger, Limited
Seagate Technology
Sharp Electronics Corporation
Silent Systems, Inc.
Sonnetech, Limited
S3, Inc.
SystemSoft Corporation
Toshiba Corporation
Trace Research and Development Center
Unisys Corporation
US Robotics
VideoLogic Limited
Vsis, Inc.
Western Digital Corporation
Yamaha Corporation
Ye Data, Inc.

PART 1

System Design Issues

C H A P T E R 1

PC 98 Design Issues

"PC 98" is a collection of PC system definitions and bus and device design requirements for 1998–99. This chapter summarizes the design goals and issues for PC 98.

Contents

PC 98 Goals

The goals for PC 98 include the following:

- Advancing the quality of PC hardware, firmware, and device drivers by driving PC hardware platform initiatives and technical capabilities to maximize customer benefit, satisfaction, and ease of use. This will result in greater customer satisfaction and lower cost of support.
- Ensuring the availability of high-quality hardware, including hardware that supports advanced Windows 98 and Windows NT features while also ensuring the availability of lower-cost PCs that best run Windows 98 and Windows NT.
- Encouraging innovation so manufacturers and designers can pursue new design solutions and make advances for hardware. PC 98 enables new uses and new users by advancing new platform types and new usage models.

To this end, the PC 98 guidelines refer to existing industry standards or specify performance goals or benchmarks rather than prescribing fixed hardware implementations. Where this is not possible (for example, for CPU and RAM requirements), it is because acceptable benchmarks were not yet available. When appropriate benchmarks and tests are available, these will be incorporated in the design guidelines.

Selection of guidelines included in this guide was based on an evaluation of features to determine how the requirements and recommendations would support PC 98 goals. Two additional considerations governed selection:

- Clarification of system support or design related to Windows 98 and Windows NT operating system architecture.
- Assurance of driver quality for both Windows 98 and Windows NT.

Basic PC 98 System Types

The design requirements and recommendations in this document provide guidance for building systems that run the Windows family of operating systems. The system type designated as “Basic PC 98” is not a set of definitions for the minimum, optimal, or best hardware required to run Windows 98 or Windows NT. Instead, this specification describes the hardware that will deliver to end users an optimal level of performance when running Windows-based applications under Windows 98 or Windows NT.

Basic PC 98 Consumer and Office Issues

The Basic PC 98 system type describes a mass-market category. The vast majority of PCs now in manufacturers’ engineering plans will be able to meet the basic requirements defined in this guide. The basic levels of performance and capabilities specified for the Basic PC 98 system type should be easy to obtain.

These guidelines are open-ended. This specification provides a starting feature set that encourages differentiation among hardware manufacturers and among product lines by adding advanced features and innovative implementations.

In this guide, a new distinction is drawn between a Basic PC designed for use in a home environment and a Basic PC designed for use in a corporate environment. These distinctions are categorized as Consumer PC 98 and Office PC 98, respectively.

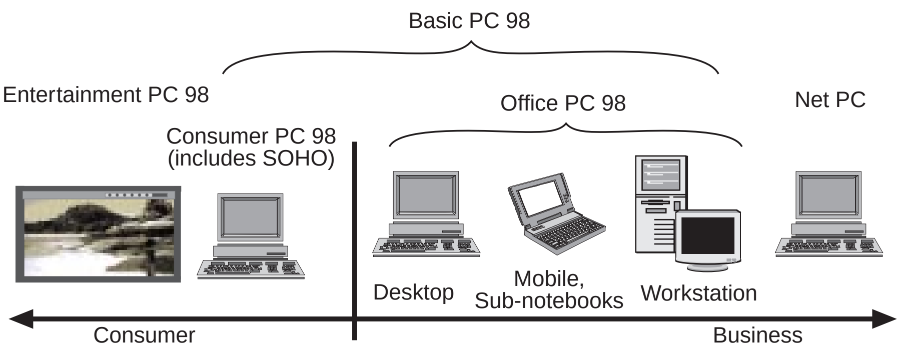
It is expected that a single OEM system will meet the Basic PC 98 specification and that a separate model number will add the “Consumer” or “Office” characteristics. Each of these categories is a variation on Basic PC 98, and each is based on the following characteristics:

- **Consumer PC.** Consumer PC 98 provides design guidelines for PCs designed for non-networked environments. This includes PCs intended for the home market in support of entertainment and game playing, plus PCs intended for Small Office/Home Office (SOHO) markets. Whatever the destination market, the Consumer PC 98 contains essentially the same hardware features and technologies, representing a baseline that can support the performance requirements for either market. The true differentiation between PCs intended for either entertainment or SOHO use is the software included with the system.

  Consumer PC 98 also comes equipped for immediate connection to the Internet using dial-up communications. To support running games, graphical applications, and entertainment and educational media titles, Consumer PC 98 includes higher quality graphics capabilities than are required for Office PC 98.

- **Office PC.** The greatest differentiating feature of Office PC 98 is that it supports PC 98 requirements intended to reduce TCO in the corporate environment, including support for an upgradable BIOS and remote boot capabilities.

  Office PC 98 is designed to run productivity applications, particularly in a network environment. Office PC 98 comes equipped with a network adapter.



The following tables summarize the basic components for each system type.

Consumer PC 98 System Summary

PC 98 reference	Required	Recommended
System requirements "Basic PC 98"	Basic PC 98 minimum, including complete OnNow support	—
System buses "USB"; "IEEE 1394"; "PCI"	Basic PC 98 minimum (USB and PCI) No Industry Standard Architecture (ISA) add-on devices CardBus for mobile PCs	IEEE 1394 Device Bay
I/O devices "I/O Ports and Devices"	Basic PC 98 minimum	Universal Serial Bus (USB) game pad Devices use external bus Remote control
Graphics and video components "Graphics Adapters"; "Video Components"	Hardware support for 2-D graphics acceleration and some 3-D acceleration features Driver support for 3-D software acceleration	3-D hardware acceleration Accelerated Graphics Port (AGP) Video port Analog television tuner Television output
Audio components "Audio Components"	—	PC 98 audio Digital ready Support music synthesis
Communications "Modems"	Internal 33.6-Kbps V.34-1996 modem	High-speed dial-up link with Network Driver Interface Specification, Version 5.0 (NDIS 5.0), driver support
Storage components "Storage and Related Peripherals"	Bus mastering 8x CD-ROM	DVD-ROM with DVD-Video playback capabilities

Office PC 98 System Summary

PC 98 reference	Required	Recommended
System requirements "Basic PC 98"	Basic PC 98 minimum, including complete OnNow support Manageability Baseline	—
System buses "USB"; "IEEE 1394"; "PCI"	Basic PC 98 minimum (USB and PCI) No ISA add-on devices CardBus for mobile PCs	IEEE 1394 Device Bay
I/O devices "I/O Ports and Devices"	Basic PC 98 minimum	Devices use external bus
Graphics and video components "Graphics Adapters"; "Video Components"	Driver support for 3-D software acceleration	2-D and 3-D hardware acceleration AGP Television output Video port PC 98 DVD-Video and MPEG-2 playback
Audio components "Audio Components"	—	PC 98 audio
Communications "Modems"; "Networking Communications"	Network adapter with NDIS 5.0 driver	Internal 33.6-Kbps V.34-1996 modem or high-speed dial-up link
Storage components "Storage and Related Peripherals"	Bus mastering	8x CD-ROM or DVD-ROM

Entertainment PC 98 Design Issues

The *PC 97 Hardware Design Guide* introduced the Entertainment PC as a distinct category of Windows-based PC, differentiated from the Basic PC by its ease of use and the breadth and quality of its multimedia capabilities. For example, the graphics, video, and audio subsystems for Entertainment PCs are designed to optimize the capabilities of software that uses Microsoft DirectX® interfaces.

Since its introduction, nearly all key Entertainment PC technologies and design points have either appeared in innovative new products or progressed technically toward realization. Some key technologies have been adopted and advanced faster than expected.

Entertainment PC 98 aims to accelerate this momentum by appropriately evolving key system components, as well as by adding some new technologies that will support feature and performance demands that will be placed on the PC hardware industry by consumers and software developers in 1998. Compared to last year's model, Entertainment PC 98 will run faster, will deliver new types of high-quality audio, video, and data content from a wider variety of sources, will better enable the PC Theater "10-foot" usage model, and will maximize the quality of user experience for multimedia applications.

It is hoped that the Entertainment PC-8 guidelines will enable and promote innovation within the PC market by defining PCs targeted at high-quality entertainment and communications experiences along with ease-of-use features that will overcome barriers to adoption for new users. An Entertainment PC 98 system is optimized for the following uses:

- **Games.** The best titles, with the most complex, realistic graphics and audio.
- **Education.** The most engaging titles, with full-screen video, interactive animation, and so on.
- **Active Internet.** Enhanced web communications capabilities, with personalized and animated web sites, chat rooms, and so on.
- **Personal communications.** Multimedia e-mail, Internet audio phone, video phone, and so on.
- **Interactive, high-resolution television and movie viewing.** Higher video quality, real-time links to content producers, and so on.
- **Connection with traditional consumer-electronics devices.** Home theater surround audio such as Dolby Digital (AC-3) for games and DVD movies, and fast and easy capture, editing, and playback of personal video.

Following are the key design challenges for Entertainment PC 98:

- Combining a high-performance 2-D and 3-D graphics subsystem designed for the best games with better-than-television quality, full-screen, MPEG-2 motion video to deliver DVD movies, digital television (DTV), and so on.
- Enabling connection to large-screen displays, including standard televisions, for a more realistic graphics experience than smaller desktop monitors can provide.
- Implementing a high-fidelity audio subsystem on par with consumer stereo systems, enabling delivery of rich content such as games with positional 3-D audio, professionally mastered music CDs, and so on.
- Enabling new types of "digital pipes" into the PC for video, audio, and information data, such as analog and DTV signals from broadcast, cable, and satellite links.
- Enabling PC connections, using USB and IEEE 1394, to consumer-electronic devices such as camcorders, VCRs, and home-theater stereo systems.
- Providing home appliance usability for ease of use, in both desktop ("2-foot") and family room ("10-foot") usage models.
- Extending human input device support with remote control, game input controls, and other control devices that use USB, IEEE 1394, and other external connections.
- Bringing advanced but easy-to-use communications capabilities to the home, and integrating these with entertainment functions. Examples include caller ID, family-room speaker phone and video phone, and so on.

It is intended that the Entertainment PC 98 guidelines provide much room for innovation, such that OEMs will design to a variety of form factors. In addition to traditional desktop multimedia PC designs, Entertainment PCs will be designed for the PC Theater category. PC Theater systems are fundamentally targeted at a 10-foot usage model and contain the necessary hardware, software, and industrial design to enable users to enjoy PC-enhanced entertainment content in the family-room environment. It should be noted that although the Entertainment PC 98 guidelines attempt to call out some of the key design differences related to 2-foot and 10-foot usage models, it is not intended as a comprehensive specification of PC Theater design issues.

Whether defined for use on the desktop or in the family room, Entertainment PC 98 guidelines are defined to deliver the best digital home entertainment of any platform or combination of devices. The following table summarizes the Entertainment PC 98 system components.

Entertainment PC 98 System Summary

PC 98 reference	Required	Recommended
System requirements "Basic PC 98"	Basic PC 98 minimum, including complete OnNow support	—
System buses "USB"; "IEEE 1394"; "PCI"	Basic PC 98 minimum No user-accessible ISA expansion slots Two USB ports, both easily accessible Two IEEE 1394 ports, one easily accessible	Device Bay
I/O devices "I/O Ports and Devices"	Basic PC 98 minimum All input devices are Human Interface Device–compliant (HID-compliant) and use an external bus interface Game pad or joystick included	Remote control supporting standard button requirements
Graphics components "Graphics Adapters"; "Monitors"	Full AGP implementation 2-D and 3-D hardware acceleration	Large-screen color monitor Television output
Video and broadcast components "Video and Broadcast Components"	PC 98 DVD-Video and MPEG-2 playback Analog video input and capture Analog television tuner DTV support	Digital broadcast or satellite television support
Audio components "Audio Components"	PC 98 audio Support 3-D audio, independent sample rates for input and output, and music synthesis	Digital ready
Communications "Modems"	Internal pulse-coded modulation (PCM) modem	High-speed dial-up link with NDIS 5.0 support
Storage components "Storage and Related Peripherals"	Bus mastering DVD-ROM drive and DVD-Video playback	External IDE devices use IEEE 1394

PC 98 Design Issues and Compliance Dates

The requirements in this guide are intended to apply to PC systems and peripherals designed for delivery in the fourth quarter of 1998, which means that compliance testing for these guidelines will begin July 1, 1998.

This guide specifies requirements that represent only incremental changes to existing silicon designs, not changes that require a new design cycle. Most of these changes are “choice of supplier” questions, or clarify and strengthen BIOS and driver implementation details first raised in *PC 97 Hardware Design Guide*.

Compliance testing for any PC 98 guidelines that require significant hardware design changes will begin January 1, 1999.

Legacy Migration Road Map

ISA-based devices remain as the most troubling area for support issues related to PC configuration. The most common causes for PC customer support calls are resource conflicts or loss of functionality occurring when end users install ISA devices such as modems, audio, or multimedia devices.

To reduce the barriers to satisfactory user experience and to reduce the cost of customer support, designers must migrate away from incorporating ISA devices in PC systems designed for Windows 98 and Windows NT. Therefore, the most dramatic PC 98 requirements emphasize the movement away from dependency on ISA and other legacy devices.

Although these requirements might seem radical, the discussion of migration is not new. In fact, migration away from legacy devices is a fundamental issue for all hardware guidelines, beginning with the migration toward Plug and Play devices.

Processor performance has improved steadily since the ISA bus was first defined; today’s processor is executing more than 300 times faster than the original PC. Even with the advanced multitasking implementations of the Windows family of operating systems, there is still a performance impact when a very fast processor has to access a relatively slow bus such as ISA. Hardware migration away from ISA, coupled with improved software techniques, will result in an improved user experience.

Every system and device manufacturer is aware of the support burden related to configuration issues for legacy devices. Often, when users add new systems to their PCs, they upset the delicate balance of system resources assigned to accommodate a collection of ISA and proprietary device implementations. These users think they are simply installing a new application or a new modem, but they lose functionality elsewhere in their systems and only have customer support to turn to in resolving the resulting problems.

To improve system performance, reduce customer support costs, and ensure true ease of use in PC systems and peripherals, manufacturers must plan to migrate all components in their systems away from ISA and legacy devices.

Of course, this migration has to be undertaken on a step-by-step basis. The first step is elimination of non-Plug and Play devices from new systems. With the addition of ACPI support and Zero Administration features in the Windows 98 and Windows NT operating systems, Microsoft is providing a foundation in the operating system for configuration, power management, and central administration of systems and devices.

Intel and Microsoft are aware that most system-board designs for 1998 include the ISA bus, and that COM and LPT ports will be included in systems for legacy device support. However, under the PC 98 guidelines, the system must not ship with add-on devices that use these ports or the ISA bus. The ultimate goal for future designs is the complete elimination of the ISA bus.

The following list summarizes the planned migration road map:

- PC 98 systems are designed and shipped without ISA-based add-on devices. If system-board devices such as BIOS ROM, Super I/O, audio, and keyboard controller are included, then each device must meet Plug and Play design specifications either as an ACPI device object (the preferred implementation) or as an ISA Plug and Play device.

  For Entertainment PC 98, the requirement is the same as for Network PC: No ISA expansion slots shall be exposed for end-user access.
- For PC 98, printers are the only devices that can use COM or LPT ports. Manufacturers are encouraged to prepare for migration toward USB and IEEE 1394, which are the recommended implementations for PC 98 printers.
- IDE and legacy AT Attachment Packet Interface (ATAPI) devices should migrate toward IEEE 1394.
- Modems, scanners, and other input and imaging devices must not use legacy buses. USB is recommended for modems, and SCSI or IEEE 1394 is recommended as the default I/O bus for scanners and other imaging devices.
- Multiple solutions are immediately available for audio and for PC 98, the most common being PCI and USB.
- No graphics can be implemented on ISA (a PC 97 requirement). Also, higher performance graphics should migrate away from PCI slave devices and toward PCI master devices, and then on to AGP.

PC 98 allows OEMs to continue to provide legacy mouse devices and keyboards, but encourages use of USB solutions. Legacy and proprietary solutions for game devices are not compliant with PC 98 requirements.

C H A P T E R 2

PC 98 Design Initiatives

This chapter presents additional information about the key PC 98 design initiatives. Complete references for specifications and implementations discussed in this chapter are presented in the “Basic PC 98” chapter in Part 2 of this guide.

Contents

OnNow and ACPI for PC 98

The OnNow design initiative is a comprehensive, system-wide approach to system and device power control based on a group of new specifications. OnNow is the term for a PC that is always on and responds immediately to user requests or other events, but it appears to be off when not in use.

Since *PC 97 Hardware Design Guide* was published, the following industry advances have been made on the OnNow design initiative:

- Completion of *Advanced Configuration and Power Management Interface Specification, Revision 1.0.* The ACPI specification defines a new cross-platform interface to the system board. This interface enables the operating system to implement operating system–directed power management and Plug and Play on a broad array of PCs, including servers, business systems, and consumer PCs. ACPI allows system manufacturers to build systems consistent with the OnNow design initiative for instantly available PCs.

  Intel and other system-board manufacturers are now implementing the ACPI specification, and OEMs are beginning large-scale manufacturing of ACPI systems.
- Completion of bus power management specifications and device class power management reference specifications for most device classes.

  Specifications have been made widely available in the industry, and manufacturers are beginning to implement OnNow-based power management capabilities in their product lines.
- Implementation of ACPI support in operating systems, enabling a generic system-event mechanism for Plug and Play, configuration control, and power management.

  Microsoft provides full driver-level support for the ACPI specification in Windows 98 and Windows NT 5.0 operating systems. Using the assembler, debugger, and compatibility testing tools provided by Microsoft, system manufacturers have been able to design, develop, and test ACPI chip sets, firmware, and system boards.
- Implementation of OnNow power management policy plus device driver and application interfaces so that device designers and driver writers can create and test OnNow-capable devices and peripherals.

The key design progress for OnNow and ACPI focuses on the following capabilities:

- Migration of system configuration from the Plug and Play BIOS to ACPI. ACPI leverages the Plug and Play BIOS data structures in a way that is compatible with both Windows 98 and Windows NT 5.0 but independent of processor architecture implementations.
- Migration of legacy power management from BIOS Advanced Power Management (APM) 1.2 to ACPI for Windows 98 and Windows NT 5.0.
- Clarification of design issues and requirements for hardware and software in support of the OnNow-capable PC.

Current information about specifications and progress for this initiative, including details for technical implementations, can be found on the web site at http://www.microsoft.com/hwdev/onnow.htm.

Win32 Driver Model

The Win32® Driver Model (WDM) is designed to allow binary compatibility for targeted device classes written for Windows 98 and Windows NT 5.0 and future versions of these operating systems. For bus and device classes with WDM support, driver developers write only small minidrivers to expose device-specific features and can maintain one driver source-code library for minidrivers that run on both Windows 98 and Windows NT.

The WDM core provided by Microsoft for Windows 98 and Windows NT 5.0 is a subset of Windows NT kernel services, with new cross-platform application programming interfaces (APIs) for Plug and Play and power management. For each bus class and device class with WDM-based support, Microsoft provides a class driver, which is a device abstraction for a particular class of devices.

Microsoft provides the WDM core services, which are documented in the Windows NT 5.0 DDK. WDM support for Windows 98 and Windows NT includes the following:

- USB and IEEE 1394 buses
- HID-compliant devices
- WDM digital audio
- Still and video imaging
- DVD decoding
- WDM modems

Key support for many devices relies on the WDM Stream class driver, which optimizes data flow in the operating system kernel.

For related PC 98 requirements, see the driver requirements for the specific bus and device class that includes WDM support in Windows 98 and Windows NT 5.0. Implementation details are provided in the Windows NT 5.0 DDK.

Manageability Initiatives

The purpose of the manageability initiatives described in this guide is to help plan, deploy, proactively maintain, and centrally control a distributed computing environment in order to reduce the overall cost of owning and managing computers. To do this, management technology must bring together information from different technology disciplines to provide services oriented toward management functions, which can in turn decrease TCO.

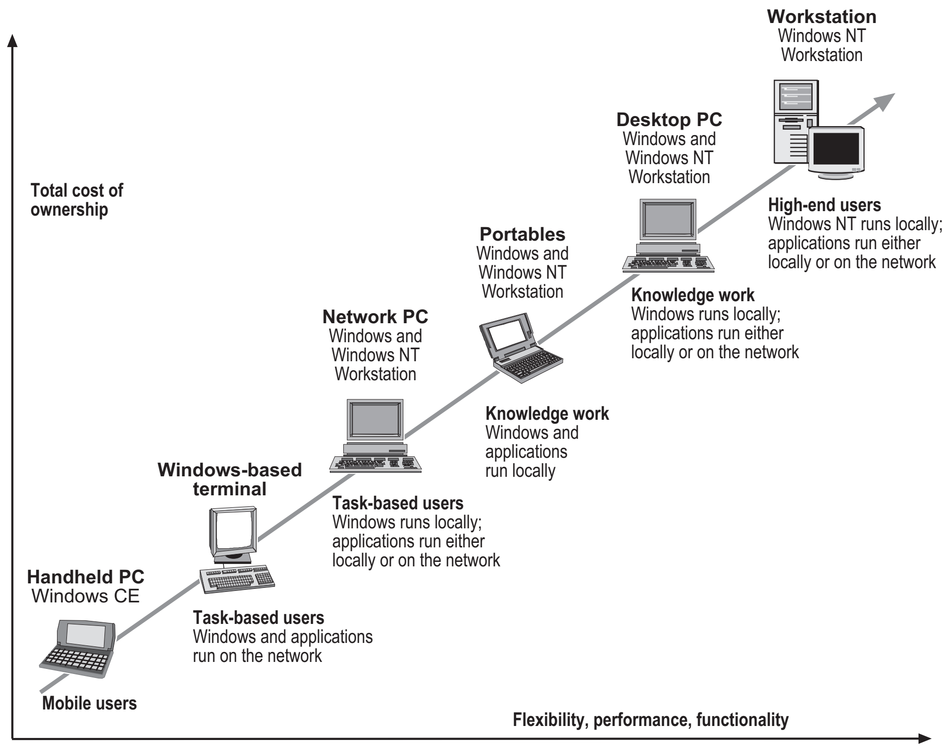
To succeed in significantly reducing TCO, management solutions must adapt to the needs and tasks of the environment to be managed. The solutions must therefore be open, flexible, and extensible: They need to support new technologies and integrate management functions supplied by more than one vendor. Such systems must conform to appropriate existing standards and have sufficient flexibility to extend support to emerging standards and technologies.

Providing management solutions requires establishing a management infrastructure in the operating system, exposing this infrastructure, and then building the tools to use it. This includes:

- Providing instrumentation as the infrastructure for manageability.
- Supporting management tools.
- Supporting new developments such as policy management in Windows 98 and Windows NT 5.0.
- Providing interfaces for enterprise management vendors.

For hardware platform designers, the technology used for platform instrumentation is of direct interest because it is a design element for their systems. Some 1997 PC platforms were instrumented with the Desktop Management Interface (DMI), as described in the *Network PC System Design Guidelines* (attached as Appendix E in the References part of this guide).

In the PC 98 time frame, a new Windows Management Instrumentation (WMI) component becomes a requirement when it is implemented in the Windows 98 and Windows NT 5.0 operating systems.



A related initiative, the Network PC (Net PC) design initiative, defines the reference design for the Net PC, a new member of the PC family that uses Intel Architecture or other microprocessor architectures and that runs the Windows 98 or Windows NT Workstation operating systems.

The Net PC is designed to reduce the cost of business computing by optimizing the design for users who do not require the flexibility and expandability of the traditional PC, and by allowing organizations to centrally manage their information technology. Although the types of business users will vary, the Net PC will be ideally suited for those involved in activities such as data entry, transaction processing, and intranet and Internet access.

The current version of the defining specification, *Network PC System Design Guidelines*, can be downloaded from http://www.microsoft.com/hwdev/netpc.htm and http://developer.intel.com. Version 1.0b is included as Appendix E in the References part of this guide.

There are several high-level design goals for Net PC, including:

- Reduce TCO in the corporate environment by designing a PC that is completely optimized for lowest TCO—centrally controllable and managed, and completely capable of being enumerated and configured through software.
- Work with the Zero Administration initiative for Windows. For more information on this initiative, see http://www.microsoft.com/management/.
- Build on existing investments in Windows-based PC computing for the corporate environment.
- Support the broadest range of applications.
- Maximize corporate return on investment in the corporate computing infrastructure.

Because the Net PC is a catalyst for improving TCO across all platforms, these guidelines are referred to in the PC 98 requirements as background information for Office PC 98. The following summarizes key Net PC design requirements:

- Completely deterministic hardware. All devices can be detected and configured by software—consequently, no ISA devices.
- OnNow-enabled, including ACPI.
- Remote operating system installation capabilities.
- No end-user accessible internal expansion capabilities (“sealed case”).
- Platform management instrumentation based on Windows hardware instrumentation for supporting operating systems such as Windows 98 and Windows NT 5.0 (based on DMI for operating systems that do not support Windows hardware instrumentation).

Notice that references to the Net PC requirements are made in this guide only to provide a context for certain Office PC 98 requirements. The actual Net PC requirements are defined in *Network PC System Design Guidelines, Version 1.0b* or higher.

Device Bay and Modular PC Design

Device Bay is a technology that enables adding and upgrading peripheral devices without opening the chassis and without turning off or rebooting the PC. Device Bay also enables peripheral devices to be easily swapped between platforms.

The *Device Bay Interface Specification* is an industry specification co-authored, jointly owned, and managed by the Compaq Computer Corporation, Intel Corporation, and Microsoft Corporation. This introduction and overview to Device Bay is based on Draft Revision 0.79, dated June 30, 1997. Availability of the Device Bay specification will be announced on the web site at http://www.device-bay.org.

The Device Bay specification defines an architecture that supports hot swapping of devices and interoperability of peripherals and platforms. A bay can be built into the chassis of any PC system that meets the operating system requirements plus all the requirements for connector receptacle, bus interface, mechanical form factor, connector receptacle, power and thermal, and controller logic as defined for bays in the Device Bay specification.

The bus interface requirement is crucial. Device Bay devices must use one or both of the industry-standard extensible bus interfaces: IEEE 1394 or USB. These buses provide a broad range of bandwidths and scalable performance to support the requirements of PC peripherals for at least the next five years.

Device Bay Device Categories

Device Bay provides manageability and interoperability for a range of PC peripherals and PC categories, including business and consumer desktop computers and portable computers, Net PC systems, and home-theater technology.

The Device Bay technologies support devices for mass storage, security, and communications and connectivity, and a variety of other devices. Device Bay technology allows OEMs, retailers, and end users to easily add peripherals in order to support specific application needs. For example, an IEEE 1394 hard drive could be added to provide a large storage medium for digital imaging or audio authoring, a DVD-ROM drive could be added to enable DVD-Video playback, or a smart card reader could be added for secure online banking or shopping.

Device Bay technologies also support swapping a hard disk drive—and thus a set of data and applications—between a desktop system and a laptop PC. And in the corporate environment, a hard disk drive could be removed from a failed system and inserted into a working system, minimizing employee downtime and thus lowering TCO.

Storage Devices. Storage is currently near the top of users' shopping lists for hardware upgrades. Users want a means of transporting large files. Device Bay supports both removable and fixed-media storage devices, including hard disk drives, tape backup, high-density floppy disk drives, CD-ROM, DVD-ROM, DVD-RAM, magneto-optical, and other removable media devices.

Communications and Connectivity Devices. Modem cards are also among the most popular end-user PC upgrades. Device Bay supports this device class, including AT-style modems, Integrated Service Digital Network (ISDN) adapters, network adapters, cable interfaces, and wireless infrared (IR) and radio frequency (RF) devices. Device Bay supports multiple-usage models and connections using USB and IEEE 1394 buses.

Security Devices. The current lack of security has been cited by consumers as a primary reason why they do not make online purchases over the Internet. Device Bay supports security devices that provide a means for user authentication. For home users, a smart card reader can enable secure credit-card shopping over the Internet. Higher levels of security can be achieved with encryption security devices. Adding a Device Bay smart card reader to a corporate or mobile system also provides a high level of data security, meeting requirements for PCs purchased by the U.S. government.

Device Bay Benefits

This section summarizes the potential benefits that Device Bay offers to end users and hardware manufacturers.

Benefits of Device Bay for End Users. The following list shows some of the ways Device Bay features can benefit end users:

- Device Bay bays are accessible. Devices can be inserted without opening the case.
- Devices inserted into Device Bay-capable bays are automatically configured, and devices can be inserted into, removed from, and swapped among Device Bay-capable bays while the PC is powered up.
- Fear of obsolescence is reduced because future higher performance peripherals will work in Device Bay bays and, as time goes on, earlier Device Bay peripherals will be accepted by future Device Bay-enabled systems.
- A broad diversity of peripherals can be Device Bay devices, and PCs can be quickly and easily reconfigured to a user's specific application needs at any given time.

Benefits of Device Bay for OEMs. The following list shows some of the ways Device Bay features can benefit OEMs:

- Device Bay's modular nature allows for cost-effective delivery of tailored PC configurations. Device Bay simplifies the manufacturing process and system configuration, allowing OEMs to design and deliver customized systems while optimizing manufacturing and distribution processes.
- Device Bay enables simplified product design and rapid adoption of new technologies into product lines, without altering either PC system design or manufacturing processes, because of standardized Device Bay form factors and device interfaces.
- Device Bay reduces obsolescence issues for OEMs.
- Device Bay lowers support costs and TCO. A principal goal of the Device Bay specification is to reduce support costs related to improper installation of new peripherals. Device Bay design ensures that many current configuration conflicts will be eliminated, thus reducing support calls.

Benefits of Device Bay for IHVs. The following list shows some of the ways Device Bay features can benefit independent hardware vendors (IHVs):

- Device Bay enables development of new product segments and enables faster integration of devices by OEMs into their platforms. Implementing new designs based on Device Bay will also mean more rapid adoption of new technologies once an installed base of Device Bay-enabled systems is present.
- Device Bay specifies a standardized design for device interfaces, connectors, and form factors. The enhanced interoperability of Device Bay ensures that IHVs have a clear indication of what to build and that they can realize great economies of scale for connectors, casings, and other components.

PC 98 and Device Bay

Device Bay is recommended for PC 98 systems. The following features are required to implement Device Bay in a PC system design:

- One USB and one IEEE 1394 port for each Device Bay-capable bay in the system, power for the bay (compliant with the Device Bay specification), and a controller for the bay (Device Bay Controller, compliant with the Device Bay specification).
- Peripherals that interface with either the USB bus, the IEEE 1394 bus, or both, and that support relevant USB device class specifications.

PC 98 compliance testing for Device Bay is expected to begin January 1, 1999, subject to availability of hardware components. For complete requirements, see the "Basic PC 98" chapter in Part 2 of this guide. See also the respective chapters for USB, IEEE 1394, and specific peripheral device requirements.

DirectX and DirectShow for Windows and Windows NT

The Microsoft DirectX foundation provides low-latency interfaces to media hardware. Previously, the primary market focus for these technologies was entertainment titles, but these APIs also provide a solid foundation for the media services required for Internet applications. They also provide the media foundation for a broad range of productivity applications, enabling high-performance media with hardware acceleration.

Microsoft DirectDraw® is the Windows system component that allows direct manipulation of video display memory, hardware block transfers (bit-blters), hardware overlays, and page flipping. DirectDraw performs the common functions required by both hardware and software emulation implementations while maintaining compatibility with the Windows Graphics Device Interface (GDI). This provides compatibility with existing Windows applications and device drivers. The user will experience the highest quality performance when using new hardware that provides built-in DirectDraw acceleration and rendering capabilities.

Direct3D® is a DirectX technology that provides access to hardware acceleration for 3-D rendering. Some basic and general 3-D capabilities will become pervasive in entertainment software by the end of 1998. These capabilities should be provided in all graphic cards to improve the performance of 3-D games, business graphics, Internet 3-D file viewing (virtual reality modeling language, or VRML), and professional 3-D applications.

DirectShow™ (formerly known as ActiveMovie™) provides access to hardware acceleration for MPEG-1 playback, which will become increasingly important for high-performance video in the context of games, Internet content viewing, computer-based training, and desktop video conferencing. Some PC 98 hardware requirements ensure support for video playback on all PCs running Windows operating systems.

DirectSound® provides a low-level and high-performance audio API, including 3-D sound spacialization (DirectSound3D) and MIDI (DirectMusic™) APIs.

DirectInput® provides a low-level and high-performance input device API to support keyboards, mouse devices, joysticks, and so on. DirectPlay® provides a collaborative communications layer.

For PC 98, all related drivers must support relevant DirectX capabilities, including DirectDraw Video Port Extensions (VPE) for graphics device drivers.

Other Design Initiatives for PC 98 Hardware

The following summarizes the important design directions and issues for PC 98.

Graphics Adapters. Requirements for 2-D and 3-D acceleration are refined for PC 98, and new requirements are introduced for supporting a video port and DirectDraw VPE for improved display of video playback.

Video, Still Image, and Broadcast Capabilities. New support and capabilities in the operating system, including kernel-streaming support implemented under WDM as well as hardware advances in the industry, have influenced changes in the hardware design requirements for video and still-image components. Principally, PC 98 guidelines for video technologies clarify requirements for MPEG-2 playback and define performance requirements for data transfer and playback quality. New requirements are defined for digital broadcast or satellite television.

Audio. The PC 98 requirements specify guidelines for digital-output readiness and new performance metrics.

Storage. The PC 98 requirements complete the migration to bus master support for all controllers and devices. Other changes clarify implementation requirements for DVD, again emphasizing the importance of bus mastering. The migration away from ISA and toward IEEE 1394 for storage is emphasized in the design recommendations.

Modems. New requirements, including new guidelines for controllerless and software modems, are defined for PC 98 so as to stay abreast with changes in the current specifications and in the modem industry. An important design issue is the emphasis on migration away from ISA.

Networking Communications. All network devices, including network interface cards, Asynchronous Transfer Mode (ATM), and ISDN implemented as a network adapter, must be implemented with NDIS 5.0 drivers and new Windows-style INF files. In addition, all network adapters must have performance capabilities for filtering multiple multicast addresses in order to support new “push” technologies for applications such as Microsoft Internet Explorer 4.0. For a PC 98 system that uses a network adapter as a boot device to support installing the operating system, the network adapter must be compatible with remote new system setup capabilities defined for manageability.

Complete details about the design changes and new requirements are provided in Part 4 of this guide.

P A R T 2

PC 98 Systems

C H A P T E R 3

Basic PC 98

This chapter provides a summary of features for the system types defined in the PC 98 requirements. As introduced in Part 1 of this guide, this chapter describes two different forms of the Basic PC 98 system. It is assumed that the OEM designs and builds a baseline platform that meets the Basic PC 98 requirements, with specific models targeted for either of two categories: Office or Consumer.

For definitions of common terms, acronyms, and abbreviations used in this guide, see the Hardware Glossary in the References part of this guide. See also the "Conventions Used in This Guide" section in the Welcome part of this guide.

Important: The system requirements defined in this guide provide guidelines for designing PC systems that will result in an optimal user experience with typical Windows-based applications running under either the Microsoft Windows or Windows NT Workstation operating systems. These design requirements are not the basic system requirements for running the Windows operating system.

Contents

Note: All references in this chapter to requirements for Network PC (Net PC) systems refer to similarly named requirements in the *Network PC System Design Guidelines, Version 1.0b,* co-authored by Compaq Computer Corporation, Dell Computer Corporation, Hewlett-Packard Company, Intel Corporation, and Microsoft Corporation. The Net PC guide is included as Appendix E in the References part of this guide. Information about Net PC is provided as a context for Office PC 98 requirements; Net PC is not a PC 98 system type.

Basic PC 98 General System Requirements

This section presents a summary of the general system requirements and recommendations, including system board, memory, and BIOS requirements.

1. System performance meets PC 98 minimum requirements

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

The performance requirements for PC 98 systems are based on the minimum computational capabilities and performance necessary to support the demands of Windows-based applications. The increased requirements over those defined in *PC 97 Hardware Design Guide* represent the estimated processing demand and processing capability of the lowest-end processor by mid-1998.

For systems based on Intel Architecture processors, the minimum PC 98 performance requirement consists of the following:

- 200-MHz Pentium processor with Intel MMX technology, or equivalent performance.

  The minimum microprocessor capability is specified to support the demands of rich media, Internet access, and conferencing. The performance requirement for media enhancement is specified to ensure that the system meets performance targets at minimum platform power.
- Minimum 256K Level 2 (L2) cache, or equivalent performance.
- 32 MB minimum system memory.

  Recommended: 64 MB. Memory should be 66-MHz Dynamic Random Access Memory (DRAM) or better.

  For a PC 98 system with 32 MB or more of memory, only 4 MB of memory can be locked and unavailable for the system to use at boot time. The remaining memory cannot be locked from use by the operating system. This minimum requirement does not preclude applications that use dynamically allocated memory for audio or video playback or other temporary uses.

For systems based on RISC architecture, the requirement is a Windows NT-compatible RISC processor. Notice that all PC 98 requirements for RISC-based PCs are for the Windows NT operating system only. There are no plans to enable Windows to run on RISC-based PCs.

Note: If multiprocessor support is provided in any system using Intel Architecture, such support must comply with *MultiProcessor Specification, Version 1.4* or higher, and the Advanced Programmable Interrupt Controller (APIC) extension to the ACPI 1.0 specification.

For guidelines and exceptions for other system types, see the “Workstation PC 98” and “Mobile PC 98” chapters in Part 2 of this guide.

2. System design meets ACPI 1.0 specification and PC 98 requirements

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

Recommended: Thermal model and fan control as defined in Section 12 of the ACPI 1.0 specification.

The system board must support the *Advanced Configuration and Power Interface Specification, Revision 1.0* or higher. This requirement ensures that the system correctly supports Plug and Play and power management as described in Part 1 of this guide. ACPI support must include the following:

- Power-management timer.
- Power button in compliance with the ACPI 1.0 specification. This should be implemented as described in the "Hardware design supports OnNow initiative" requirement later in this section. A separate reset switch is an acceptable alternative to the ACPI-specified override mechanism.
- Real-time clock alarm that supports wake up due to a scheduled time and day of the month. Notice that the day-of-month feature is a requirement for PC 98, although it is an optional feature in the ACPI 1.0 specification.
- The S5 (soft-off) state, as required in the ACPI 1.0 specification, plus at least one of the S1, S2, or S3 sleep states. Notice that one of sleep states S1–S3 is required for PC 98, although this is only recommended in the ACPI 1.0 specification. The S4 and S4BIOS sleep states are not sufficient to meet this requirement.

  Recommended: For maximum power savings and system response, support of the S3 sleep state is recommended.
- System control interrupt and Status and Enable (STS/EN) bits for the power-management timer, power button, and real-time clock.
- Description table for system-board devices, including host PCI bridges. This table defines the complete hierarchy, including all non-Plug and Play devices to be enumerated and all other devices for which power management or removal capabilities have been added by the system-board design.
- ACPI control methods necessary to configure each bus and device enumerated using ACPI. This is as described in the "Each bus and device meets Plug and Play specifications" requirement in the "Basic PC 98 General Device Requirements" section later in this chapter.
- USB host controller able to wake system from at least one sleeping state (S1, S2, or S3) using ACPI mechanisms. This is a requirement for PC 98, although it is only a recommendation in the ACPI 1.0 specification.
- No capabilities to disable system ACPI support using CMOS or other means.

Implementing the thermal model and fan control as defined in the ACPI 1.0 specification is recommended as a means of running the PC quietly while working, and turning the fan off while the PC is sleeping. Notice that a hardware-based open-loop thermal control is an acceptable implementation for system cooling if it meets the capabilities defined in the “Audible noise meets PC 98 requirements” item later in this chapter. However, the recommended implementation is a closed-loop control using the PC’s processor, an embedded controller, or both. If a closed-loop implementation is used, it must comply with the ACPI 1.0 specification.

Note: Any other system-board power management or Plug and Play features must be implemented in compliance with the ACPI 1.0 specification, even if a particular feature is not a specific PC 98 requirement or recommendation.

3. Hardware design supports OnNow initiative

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

Recommended: If telephone answering machine (TAM) capabilities are built into the system, then a Message Waiting indicator should be included on desktop systems.

Elements of the OnNow design initiative ensure that the operating system and device drivers control the state of individual devices and the system board. The following support is required for PC 98 hardware:

- The user experiences the PC as off when it is in a sleep state.

  At a minimum, the hard disks, CD drives, display, sound, input devices, and fans must be perceived as off while the system is in a sleep state (for example, no noise or lights other than the status indicator).

- The user can easily see whether the PC is working or sleeping.

  A non-flashing light-emitting diode (LED) sleep indicator that is a different color than the wake indicator is the preferred implementation. A slowly blinking—less than 1 Hz—LED indicator is an acceptable implementation. Notice that the nonvolatile sleep state (S4 or S4BIOS) should appear to the user as the off state; that is, these states should have the same indicator.

  At a minimum, the PC 98 system must provide one or more indicators to show whether the system is in the working or sleep state.

- The user can easily control power through switches and software.

  The system must provide an easily accessible power switch that can be controlled by software and that supports the functionality required in Section 4.7.2.2.1 of the ACPI 1.0 specification.

To meet this requirement, an OnNow PC can have either a power button or a sleep button. The recommended implementation for both desktop and mobile PCs is to have both. If both buttons are implemented, the sleep button should be the user’s primary switch interface and must be easily distinguishable from the power button. The preferred implementation is to hide the power button.

The function of these buttons is determined by the operating system. In single-button configurations, it can be used for either sleep/wake transitions (G0<–>G1/S1-S4) or off/on transitions (G0<–>G2/S5), depending on user preference and the policy set in the operating system and use of an operating system–provided user interface. The default action for the sleep button is to cause the machine to enter a sleep state. The default action for the power button is to shut down the operating system and power off the machine. In a two-button configuration that includes separate power and sleep buttons, the operating system user interface will allow only the default actions.

For PC 98, in the case of a hardware or software failure that prevents normal operation of the software-controlled buttons, the switch capabilities must include an override mechanism for turning off the PC. Notice that the override mechanism is not an alternative way for the user to turn off the PC in normal operation; it is only a fail-safe function for fault conditions.

The implementation recommended in Section 4.7.2.2.1 of the ACPI 1.0 specification is a 4-second override mechanism. The override can be on either the power button or the sleep button in a two-button configuration.

Note: It is strongly recommended that the override be associated with the sleep button in order to establish an industry-standard implementation.

An acceptable but not recommended alternative is a separate hidden or recessed switch that cannot be mistaken for either the power button or the sleep button. Notice that although the ACPI 1.0 specification suggests the override be associated with the power button, the recommended implementation for an OnNow PC is to have the primary and most accessible button be the sleep button.

Equivalent button functionality can be provided using a keyboard key. If the power switch is provided on the keyboard, the key must be clearly labeled and must consist of a single keystroke for turning on the PC. (Two keystrokes are permissible for turning off the PC.) The single keystroke ensures accessibility for persons with disabilities. For information about the correct scan codes for keyboard power switches, see http://www.microsoft.com/hwdev/pcfuture/.

This requirement for an easily accessible power switch does not preclude power-control capabilities such as closing the lid on a mobile PC.

Power management is supported for any of the following buses present on the system: PCI, USB, IEEE 1394, and PC Card. Any of these buses must support power management requirements as defined in the related bus standard in Part 3 of this guide.

- Each device supports the power management specifications for its class.

  All devices and drivers must support the D0 and D3 power states consistent with the definitions in the relevant device class power management reference specification or the *Default Device Class Power Management Specification, Version 1.0* or higher.

  This means that each device can successfully survive a system sleep/wake transition (D3 to D0) without losing functionality and without requiring user intervention to restore functionality. This applies whether or not system power is removed while the device is in the D3 power state.

  Notice that there is no power consumption requirement for devices in the D3 power state. It is strongly recommended, however, that devices implement the D3 power state such that device power consumption is reduced to near zero. This recognizes that there is no requirement to retain any device context because it will be preserved or restored by the driver when returning to the D0 power state.

  It is recommended that devices and drivers support the D1, D2, or both low-power states and also support the defined wake-up events as designated in the related device class power management reference specification.

4. BIOS meets PC 98 requirements for OnNow support

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

This requirement does not apply for RISC-based PCs, except for the requirement for fast power-on self test (POST). For PC 98, the following BIOS support is required:

- Fast POST. The system must be available to the user as fast as possible. Although a specific time limit is not established, the basic recommendation is that power on to the bootstrap loader should occur within 5 seconds, plus hard disk ready time. The following are recommended ways to reduce processing overhead to make system boot time as fast as possible:
  - No video memory test, and limited test for DRAM size.
  - No tests for serial or parallel ports.
  - No floppy disk test or media check (boot from hard disk only).
  - No tests for hard disk controller or drive type (if the system does not include swappable drives).
  - Test execution controlled using Windows-based control panel or application that can be scheduled to run periodically at off-hours.
  - Fast POST mode for BIOS that can be disabled by the user for troubleshooting.

- Minimal time for resume from sleep state.

  Resume from sleep state (S1–S4) to operating system handoff must occur within 500 ms. This requirement does not apply for the S4BIOS state. For all other sleep states, the time to operating system handoff is measured from when the processor starts running (first instruction) until the BIOS jumps to the Waking Vector in the ACPI firmware control structure table, as described in Section 5.2.6 in the ACPI 1.0 specification.

- Minimal start-up display. System start-up draws the end user's attention only in case of errors or when there is a need for user action.

  The default configuration must allow a beep during the boot process only in case of an error. The only screen display allowed is the OEM splash screen, which can include information such as copyright notices. By default, the system must be configured so the screen display does not display memory counts, device status, and so on, but presents a "clean" BIOS start-up. However, this requirement does not preclude the following:

  - Presenting a blank start-up screen.
  - Providing a hot-key override to display screen messages for trouble-shooting or to display user-definable CMOS settings.
  - Presenting text-based end-user action messages—for example, messages to display the setup hot key, system help hot key, password entry, network log on for remote booting, and so on.
  - Presenting manufacturer branding messages.
  - Providing a CMOS option to turn the clean start-up screen off and on.

The intention of this requirement is to ensure that the end user is not presented with confusing information and unnecessary visual display, and to ensure that access to error information remains available using a hot key.

5. BIOS meets PC 98 requirements for boot support

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

This requirement does not apply for RISC-based PCs. ACPI BIOS entries, as defined in Section 1.6 of the ACPI 1.0 specification, should be the same for supporting either Windows 98 or Windows NT 5.0. In general, the run-time services portion of the Plug and Play BIOS is replaced by ACPI and therefore is not required. In fact, the Plug and Play BIOS must be disabled when ACPI is enabled. The Extended System Configuration Data (ESCD) calling interface is not supported by Windows 98 or Windows NT 5.0.

The PC 98 requirements include the following:

- Preboot execution environment supported.

  For Office PC 98, the execution environment provided by the PC system must conform to the description given in "Attachment B: Preboot Execution Environment" of *Network PC System Design Guidelines.*

  For Consumer PC 98 and Entertainment PC 98, this means providing a PXENV unique system ID structure in the system BIOS or CMOS, as defined in "Attachment B: Preboot Execution Environment" of *Network PC System Design Guidelines.*

- BIOS boot for CD-ROM supported.

  For any system that includes a CD-ROM, the system BIOS or option ROM must support the No Emulation mode in *El Torito—Bootable CD-ROM Format Specification, Version 1.0,* by IBM and Phoenix Technologies, Limited, or an equivalent method that supports the Windows NT CD-ROM installation process.

- BIOS boot for network adapter supported.

  For any PC 98 system that includes a network adapter, the system BIOS must comply with the requirements defined in Sections 3 and 4 (as they apply to Plug and Play devices) of the *Compaq, Phoenix, Intel BIOS Boot Specification, Version 1.01* or higher, which describes the requirements for Initial Program Load (IPL) devices.

- Dates beyond the year 2000 correctly supported in BIOS and CMOS.

The following features are required for Office PC 98 systems to ensure manageability and security:

- Security such as a pre-boot password provided to protect enable/disable capabilities for hardware components before the operating system boots.

  This capability is required for Office PC 98 systems and is recommended for other system types. The purpose of this feature is to prevent end users from accidentally or purposefully circumventing operating system–level security and control as applied by an administrator.

- BIOS updates and revisions supported.

  This item is required for Office PC 98 and Net PC systems. BIOS updates must be implemented in order for BIOS ROMs to be upgraded to a new image through OEM-provided programs using either (1) the remote new system setup mechanism that will be downloaded and executed at boot time or (2) normal file access and execution methods when the system is fully booted into the normal operating system environment. For information about requirements related to the remote new system setup mechanism, see the "Network Communications" chapter in Part 4 of this guide.

  Recommended: Implement a mechanism to authenticate the requester of the update programming. Implement a mechanism to validate that the program arrived intact after download.

  A working group is developing a mechanism to implement this requirement for non-volatile storage update capabilities. When available, this information will be available on the web site at http://www.teleport.com/~nsispec/.

  If option ROMs are provided, they must also be capable of being upgraded.

- BIOS boot for USB keyboard supported if USB is the only keyboard.

  For any Intel Architecture system with a USB keyboard as the only keyboard in the system, the system BIOS must provide boot support as defined in *Universal Serial Bus PC Legacy Compatibility Specification, Version 0.9* or higher.

Basic PC 98 Physical Design Requirements

This section summarizes physical design requirements and recommendations for PC 98 systems. These requirements are in addition to those related to the OnNow initiative for power-state indicators and easily accessible power switches.

6. All expansion slots in the system are accessible for users to insert cards

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

The internal expansion slots cannot be physically blocked by components or devices provided with the system. This requirement does not exclude configurations that allow space only for half-height cards for some slots, for passive back planes for connectors, and so on.

For Net PC systems, user-accessible internal expansion slots are not allowed. For mobile guidelines and exceptions, see the "Mobile PC 98" chapter in Part 2 of this guide.

7. Audible noise meets PC 98 requirements

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

A PC 98 system must be “silent” in the sleep state. That is, it must be perceived as not significantly noisier than the off state to typical users, relevant to an operating position appropriate to the PC’s form factor (such as desktop, minitower, or laptop) and the ambient noise level of its normal usage environment (such as corporate office, home office, family room, and so on). This requirement applies primarily to fan noise, as all other devices will not be active in the sleep state.

It is hoped that this definition will become more objective over time through standardization of acoustic noise measurement and reporting procedures for PCs. Intel and Microsoft are working on proposals for acoustic noise measurement and reporting. The goal is to achieve common PC acoustic noise measurement methods based on established international standards. With such methods in place, end users will be able to receive reliable acoustic noise specifications about PCs similar to those available for other product categories such as automobiles and appliances.

Although this PC 98 requirement does not specify noise limits for PCs in idle and working states, manufacturers are strongly encouraged to design systems that operate as quietly as possible, especially Entertainment PCs designed for use in thehome family room. It is expected that future design guidelines will propose operational noise limits for the Entertainment PC category.

8. System and component design practices follow accessibility guidelines

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Recommended*

Accessibility design guidelines are provided in the “Accessibility” appendix in the References part of this guide. These guidelines were developed in consultation with the Trace Research and Development Center at the University of Wisconsin at Madison. This recommendation will not become a requirement.

9. Internal system modification capabilities are not accessible to end users

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Recommended*

This recommendation is based on goals to reduce TCO by ensuring that end users are prevented from inadvertently (or purposefully) altering the predefined software and hardware configurations. This recommendation, which is a requirement for Net PC systems, can encompass any of the following system requirements as defined in the *Network PC System Design Guidelines:*

- Lockable or sealed-case design, where internal expansion capabilities are not end-user accessible
- Upgrade capabilities for RAM and CPU are not end-user accessible

10. System design provides physical security

Consumer PC 98	Office PC 98	Entertainment PC 98
Recommended	*Recommended*	*Recommended*

To prevent unauthorized hardware access, the following security features are recommended for PC 98 systems and required for Net PC hardware:

- External drive devices have locking capabilities. Each removable media device should be capable of being locked to prevent unauthorized data access. This means that the device is rendered inoperable, either electronically or mechanically, when locked.
- PC case and switches have locking capabilities to prevent unauthorized internal access. An OEM-specific method can be implemented, either electronically or mechanically.

Basic PC 98 General Device Requirements

The requirements in this section apply for every device, whether present on the system board or as an expansion device provided by the OEM in a default system configuration. Most general device requirements are related to Plug and Play capabilities.

11. Each device and driver meets PC 98 device requirements

Consumer PC 98	Office PC 98	Entertainment PC 98
Required	*Required*	*Required*

Each device must comply with all PC 98 requirements for the related device class, whether the device is provided in the PC system as an expansion card or as an external device.

Drivers must be provided for both Windows and Windows NT operating systems. The manufacturer does not need to supply a driver for a device if the device passes PC 98 compliance testing using a driver provided with the operating system. Notice that not all drivers provided with Windows or Windows NT comply with the basic and device-specific requirements for drivers as defined in this guide.

In addition to the device requirements in this section, see also the specific requirements for each device class in Part 4 of this guide.

12. Each bus and device meets Plug and Play specifications

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

Each bus and device provided in a PC 98 system must meet the current Plug and Play specifications related to its class, including requirements defined in Section 6 of the ACPI 1.0 specification and clarifications published for some Plug and Play specifications. This includes requirements for automatic device configuration, resource allocation, and dynamic disable capabilities.

For information about new Plug and Play support under Windows NT 5.0, see the Windows NT 5.0 DDK.

The following shows current version numbers for all Plug and Play specifications:

- *PCI Local Bus Specification, Revision 2.1* (PCI 2.1)
- *Plug and Play External COM Device Specification, Version 1.0*
- *Plug and Play Industry Standard Architecture (ISA) Specification, Version 1.0a*
- *Clarification to Plug and Play ISA Specification, Version 1.0a*
- *Plug and Play Parallel Port Device Specification, Version 1.0b*
- *Plug and Play Small Computer System Interface Specification, Version 1.0*
- *Universal Serial Bus Specification, Version 1.0*

Plug and Play specifications for IEEE 1394 are defined in this guide. For information, see the “IEEE 1394” chapter in Part 3 of this guide.

Note: Standard system devices are excluded from this requirement. The system can reserve static resources for devices such as programmable interrupt controllers (PICs) 1 and 2, timer (8254-2), keyboard controller (8042), real-time clock, DMA page registers, DMA controllers 1 and 2, and math coprocessor. For systems based on Intel Architecture processors, these fixed resources are located at I/O addresses under 100h and can also include a Nonmaskable Interrupt (NMI). For more information, see the “Legacy Support” appendix in the References part of this guide.

In addition, systems designed to run only on Windows NT are not required to meet PC 98 requirements for legacy Plug and Play support. If the system is designed to run either Windows 98 or Windows NT, it must meet all PC 98 requirements for legacy Plug and Play support.

13. Unique Plug and Play device ID provided for each system device and add-on device

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

Each device connected to an expansion bus must be able to supply its own unique ID, as defined in the current Plug and Play specification for the bus that it uses. The following defines the specific requirements for Plug and Play device IDs:

- Each separate function or device on the system board must be separately enumerated; therefore, each must provide a device ID in the manner required for the bus it uses.
- If a device on an expansion card is enumerated by the BIOS, it must have a unique ID and its own resources according to the PC 98 device ID requirements for the bus to which the card is connected. This includes devices that are separately enumerated on multifunction cards or multifunction chips.

  CardBus devices must meet the requirements defined in the "PC Card" chapter in Part 3 of this guide.

The following are exceptions to this requirement:

- Legacy devices attached to the ISA bus on the system board do not have unique Plug and Play IDs—for example, serial ports, parallel ports, or PS/2-compatible port devices. The method for device identification is defined in the *Plug and Play ISA Specification, Version 1.0a,* and the ACPI 1.0 specification.
- Some multifunction devices (such as Super I/O) might include devices that do not have unique Plug and Play IDs or unique PCI subsystem IDs, but that are supported by drivers provided with the Windows operating system.
- A device such as a multifunction PCI device that supports a number of functions but uses only a single set of relocatable resources does not have to provide separate IDs for each function included on the device.

In addition, for Office PC 98 and Net PC systems, if an OEM uses a proprietary mechanism to assign asset or serial numbers to hardware, this information must be available to the operating system using Windows hardware instrumentation technology, as defined in the *Network PC System Design Guidelines.*

14. Option ROMs meet Plug and Play requirements

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

This requirement applies only for devices that might use option ROM on systems based on Intel Architecture processors, whether the device is present on the system board or provided through an expansion card.

Option ROMs are usually located on cards used as system boot devices. During the boot process, option ROMs initialize the boot devices, which provide the primary input, primary output, and IPL device to boot the system. However, Plug and Play option ROMs can be used to supply the Plug and Play expansion header to devices other than boot devices, enabling them to initialize both devices when the system boots.

To design an option ROM with Plug and Play capabilities, follow the requirements described in the *Plug and Play BIOS Specification, Version 1.0a,* and *Clarification to Plug and Play BIOS Specification, Version 1.0a,* which describe the Plug and Play expansion header and the interaction between the system BIOS and the option ROM. In particular, note the following points from the specifications:

- The header contains information that identifies the type of boot device connected to the expansion card. This information allows the system BIOS to prioritize the boot devices. Shadowed copies of the option ROM must also contain the Plug and Play expansion headers.
- A Plug and Play option ROM must be able to determine whether the system BIOS complies with Plug and Play. If the system ROM is not Plug and Play-compliant, the option ROM should immediately initialize the card and hook the proper interrupt as though it were a non-Plug and Play option ROM. This allows the expansion card to be used in non-Plug and Play systems.
- An option ROM can use the system BIOS run-time functions, but these functions are not available until after the POST process has completed and Int 19 has been called. In particular, an option ROM must not hook the following interrupts until the system BIOS calls the boot connection vector contained in the Plug and Play expansion header: Int 9h, Int 10h, Int 13h, Int 18h, or Int 19h. Option ROM routines must not try to use these run-time functions until that time, because the results can be unpredictable.

Option ROM requirements for specific devices are defined in the “IDE and ATAPI” and “SCSI” chapters in Part 3 of this guide and in the “Graphics Adapters” and “Storage and Related Peripherals” chapters in Part 4.

Note: Systems designed to run only on Windows NT are not required to meet PC 98 requirements for legacy Plug and Play support.

15. "PNP" vendor code used only to define a legacy device's CompatibleID

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

All legacy devices not enumerated by the system-board interface must not use "PNP" in their vendor and device codes. The PNP vendor code is reserved for Microsoft and for vendors whose hardware is specifically assigned a particular ID. Other hardware can use a PNP code only when defining a device's CompatibleID and only after first indicating the device's HardwareID in the Plug and Play header.

Use of CompatibleIDs is strongly recommended for devices that use device drivers provided with the Windows operating system, such as a Standard PC COM Port (PNP0500) or Sound Blaster 16-compatible Sound Device (PNPB003).

For information about using PNP CompatibleIDs, see the "Device Identifiers" appendix in the References part of this guide. To obtain a unique PNP vendor ID, please send a request by e-mail to pnpid@microsoft.com.

16. Device driver and installation meet PC 98 requirements

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

Each device must have drivers for both Windows and Windows NT to ensure correct support under both operating systems. For some device classes, this support can be provided using a WDM driver, as defined in the related device requirements in Part 4 of this guide.

General driver requirements. The manufacturer does not need to supply a driver for a device if the device passes PC 98 compliance testing using a driver provided with the operating system. If the manufacturer supplies a driver, the device drivers and installation requirements include the following:

- All devices and drivers must pass PC 98 compliance testing. Each device included in a PC 98 system must comply with the PC 98 requirements and must have supporting 32-bit device drivers for the CPU platform and operating system.

  Notice that for the Windows operating system, the display driver file is a Win16 module.

- All configuration settings are stored in the registry. The driver must not use INI files for configuration settings.

  The driver must also include correct provider, version, and copyright entries. This information is displayed in the user interface, such as Device Manager in Windows.

- The correct minidriver, virtual device drivers (VxDs), or any other manufacturer-supplied files specified in the device's INF file must be installed in the correct location.

  For manufacturer-provided files, the vendor must *not* be identified as Microsoft and all other copyright and version information must be correct for the manufacturer.

- Driver installation and removal must use Windows-based methods as defined in the Windows and Windows NT DDKs.

  The device driver must be able to be removed using Windows-based software, which can be managed using either the Windows Control Panel option for removing devices or its own remove utility. For information, see "Driver Installation" in the Windows NT 5.0 DDK and "Windows 95 Class Installers and Network Driver Installers" in the Windows 95 DDK.

  However, any software applications included with the device can be installed using an alternate Windows-based installation method as defined in the Win32 SDK.

  Also, any software components and registry entries installed during driver installation must be removed during driver uninstallation.

- Driver files provided by the vendor must not use the same file names as used by files included in Microsoft operating systems and provided as either retail or OEM products, unless specifically agreed upon with Microsoft.
- It must be possible for the device's driver to be installed using a mechanism, such as a script or special software, for supplying required parameters without the user being present.
- In order to ensure that the user can correctly change settings, a Windows Help file must be provided if special driver parameters are used. The device's installation routine must install the Help file as part of the setup program. The user interface for the device's dialog boxes must display the correct Help file, and the Help file must contain relevant information to assist the user. The guidelines for implementing a Help file are defined in the Windows NT DDK.

System-specific requirements. For systems that come pre-installed with either Windows 98 or Windows NT 5.0, the following requirements apply:

- For any device for which WDM-based support is provided in the operating system, the driver supplied by the manufacturer must be a WDM minidriver.
- Every driver (or minidriver) must support Plug and Play and power management I/O request packets (IRPs).

For systems that come with Windows NT pre-installed, only 32-bit protected-mode components are installed. No real-mode or 16-bit protected-mode components can be provided in order to operate under Windows NT.

For systems that come pre-installed with Windows, the following requirements apply for drivers:

- Every VxD must support Plug and Play and power management messages.
- The driver must provide power management support as required by any device class power management reference specification.
- Any real-mode components provided for backward compatibility should use separate installation procedures. Although installation of Windows-based components must not make entries in Autoexec.bat or Config.sys, the separate real-mode installation program can make such entries but must not modify the registry, Win.ini, or System.ini.

17. Minimal user interaction needed to install and configure devices

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

After physically installing the device, the user must not be required to perform any action other than to insert the disks that contain drivers and other files. The user should have to restart the system only for devices that do not support hot plugging.

As specified in the requirement, “Hot-plugging capabilities for buses and devices meet PC 98 requirements,” later in this section, devices that use USB, IEEE 1394, or PC Card must support hot plugging. For devices that use other buses, detection occurs when the system is powered on after the device is inserted.

The following requirements must be met:

- The device is immediately functional without restarting the system.

  It is acceptable to require rebooting for primary system devices such as the primary graphics adapter and the primary hard disk controller. In all cases, however, changing configuration settings must not require the end user to make jumper changes.
- Software settings are available for configuring all resources.

  All buses and devices on both the system board and all expansion cards must be capable of being configured by the operating system and by software (such as Device Manager in Windows) so that the user does not need to open the PC case to change the configuration. DIP switches on boot devices can be used for an initial power-on default state or for non-Plug and Play system compatibility, but such settings must be capable of being overridden by software configuration after power on occurs under Plug and Play operating systems.

  Note: This requirement does not apply for jumper settings used by the OEM to set CPU speed, select a keyboard, or make other basic system-related settings in the factory. This requirement applies only for settings that the end user must make to configure the hardware.

- Dynamic disable capabilities are supported for all devices.

  All devices must be capable of being automatically disabled by the system. Also, disabling the device must result in the freeing of all its resources for use by other devices.

  The following devices are exempt from this requirement: all legacy devices using the I/O range under 100h, keyboard controller, FDC, hard disk controller, video graphics array (VGA) memory and I/O addresses, and any BIOS memory ranges required for legacy boot support.

18. Connections use icons plus keyed or shrouded connectors

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

This requirement helps ensure that the end user can correctly make the physical connections required for adding a device to a system. This requirement includes the following:

- Wherever possible, keyed or shrouded connectors or other configurations should be used to prevent misconnection.

  The physical design of the connector must ensure that the user cannot mistakenly insert the connector into the wrong port. For specific requirements related to keyed connectors and cables for I/O controllers and peripherals, see the “IDE and ATAPI” and “SCSI” chapters in Part 3 of this guide.

- Icons are provided for all external connectors.

  The icons can be molded, printed, or affixed as permanent stickers (which can include text). Icons can be based on existing vendor designs or on the examples listed in the “Icons” appendix in the References part of this guide.

  PC 98 does not specifically require color coding of connectors and other cable markings, but the PC designer is free to implement color coding in order to enhance user accessibility.

Note: It is recognized that the design for legacy ports, such as the PS/2-compatible mouse and keyboard ports, analog audio and video jacks, and the microphone and speaker jacks, will not change and therefore cannot fully meet this requirement. However, icons and labels must be provided wherever possible to help the user make the correct connections.

19. Hot-plugging capabilities for buses and devices meet PC 98 requirements

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

Recommended: A locking mechanism to ensure that devices are removed only under operating system control or during sleep or off states.

To ensure reliable support for hot-plugging capabilities, the following PC 98 requirements must be met:

- Devices and buses must support hot plugging if using USB, IEEE 1394, or PC Card.

  When designed under their respective specifications, USB, IEEE 1394, and PC Card all support hot plugging. Any device designed to use any of these connections must support being added or removed while the system is fully powered on.

  The exception to this requirement is any device required for booting such as the primary graphics adapter. For information about supporting multiple graphics adapters, see the "Graphics Adapters" chapter in Part 4 of this guide.

- Hot plugging for PCI devices must use ACPI-based methods.

  Windows 98 and Windows NT 5.0 support dynamic enumeration, installation, and removal of PCI devices only if there is a supported hardware insert/remove notification mechanism.

  The notification mechanism is defined as part of the bus standard for CardBus bus controllers. For other solutions, such as those required for docking stations or other devices, the hardware insert/remove notification mechanism must be implemented as defined in Section 5.6.3 of the ACPI 1.0 specification.

  In order to properly function with the native support in the operating system, developing industry standards such as those referred to as PCI Hot Plug and Compact PCI must use ACPI-based methods for supporting hardware insertion and removal as defined in the ACPI 1.0 specification.

- Removable media must support the appropriate media status notification method to ensure that no loss of data or system failure results when such media is removed from the system.

  The media status notification requirements for CD-ROM, DVD-ROM, IDE, and ATAPI removable devices are defined in the "Basic PC 98 Storage and Related Peripherals" section later in this chapter. The Device Bay implementation guidelines are defined in the following requirement.

It is strongly recommended that designers ensure that surprise removal of any swappable device should not cause a system failure. A failure related to surprise removal of a swappable device includes any spontaneous reboot, system stall, or blue screen. At a minimum, the device driver should ensure that the PC system does not fail if the user accidentally pulls the device out of its socket. The only absolute way to ensure against system failure is to prevent surprise removals by including a locking mechanism, which is strongly recommended for PC 98 systems.

Another method of protection is to have the driver check whether its device is present when it receives certain interrupts. For example, CardBus cards share the same PCI interrupt as their socket controller, so interrupt handlers for both the card driver and socket driver are chained to the same PCI interrupt request (IRQ). To prevent a system fault after surprise removal of the CardBus card, its driver must check whether its device is still present whenever it reads a value such as 0xFFh in its status register, and then it must be able to recover gracefully when this occurs.

In all cases, for any failure that might occur, the PC system as a whole must be able to recover gracefully and report the condition to the end user.

For information related to implementation details and for additional design guidelines, see the article about hot-plugging support on the web site at http://www.microsoft.com/hwdev/devdes/.

20. Device Bay-capable bay and peripherals meet Device Bay specification

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

If implemented in a PC 98 system, Device Bay capabilities must meet the following requirements:

- A Device Bay Controller (DBC) compliant with *Device Bay Interface Specification, Version 1.0* or higher, and implemented as an ACPI device object on the system board, as defined in Section 8 of the Device Bay specification
- One USB controller and one IEEE 1394 controller to support all Device Bay-capable bays in the system
- One USB port and one IEEE 1394 port for each Device Bay-capable bay in the system

Any Device Bay peripherals provided with a PC 98 system must meet the following requirements:

- Compliant with *Device Bay Interface Specification, Version 1.0* or higher
- Interface with either the USB bus, IEEE 1394 bus, or both
- Support relevant USB device class specifications

21. Multifunction add-on devices meet PC 98 device requirements for each device

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

Multifunction add-on devices can contain more than one device. They must meet the requirements defined earlier in this section for automated software-only settings for device configuration, device drivers, and Windows-based installation. In addition, the following requirements must be met:

- Each function or device on the multifunction add-on device that is individually enumerated by the BIOS must provide a device ID for the bus it uses.
- The system must be able to separately access each logical device that is individually enumerated by the BIOS, configure the device resources independently, and disable individual devices in the event of a conflict.
- For each individually enumerated device, resource configuration requirements are the same as for an equivalent device on a separate expansion card. This means that registers cannot be shared among individually enumerated devices on a multifunction add-on device, but does not supersede device requirements among different bus classes.

The exception to this requirement is a device such as a multifunction PCI device, which supports several functions but uses only a single set of relocatable resources. When each device is not individually enumerated, there is no requirement to provide separate IDs and separate resources for each function on the device.

22. All devices support correct 16-bit decoding for I/O port addresses

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

Each device must support a unique I/O port address in the 16-bit address range. This requirement means that, at a minimum, the upper address lines (A10–A15) can be used as the device enable address, so that the device does not respond to addresses outside of the 10-bit address range. CardBus controllers and cards must meet the requirements defined in the “PC Card” chapter in Part 3 of this guide.

Devices that use less than 16-bit I/O decode create conflicts that cannot be resolved by a Plug and Play operating system. Phantom (alias) addressing is not supported by the Windows operating system and cannot be used to meet PC 98 requirements.

Notice, however, that this requirement does not apply for the three ISA auto-configuration registers used during device enumeration and configuration. The ADDRESS, WRITE_DATA, and READ_DATA registers will continue to use 12-bit decoding as described in the *ISA Plug and Play Specification, Version 1.0a.*

23. System-board devices use ISA-compatible addresses

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

This requirement does not apply for RISC-based PCs. This requirement is unchanged under the plan for migration away from ISA.

This requirement includes devices with I/O port addresses within the reserved range 0h–0xFFh. For information about legacy system I/O addresses, see the "Legacy Support" appendix in the References part of this guide.

Basic PC 98 Buses and Devices

This section defines specific requirements for buses and devices provided in a PC 98 system, in addition to the basic requirement for supporting the ACPI 1.0 specification defined earlier in this chapter.

Basic PC 98 System Buses

This section defines the general requirements for system buses. Additional requirements for specific buses are defined in Part 3 of this guide.

24. Each bus meets written specifications and PC 98 requirements

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

In the past, some bus designs did not fully implement all of the bus requirements on every expansion card connector. For PC 98, each bus used in the system must meet all the requirements for that bus as defined in Part 3 of this guide.

Each bus and device provided in a PC 98 system must also meet the current Plug and Play specifications related to its class, including requirements defined in the ACPI 1.0 specification and the clarifications published for some Plug and Play specifications. This includes requirements for automatic device configuration, resource allocation, and dynamic disable capabilities. See also the related Plug and Play requirements in the "Basic PC 98 General Device Requirements" section earlier in this chapter.

25. System includes USB with one USB port, minimum

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required, with 2 USB ports*

Recommended: 2 ports.

USB provides a bi-directional, isochronous, dynamically attachable serial interface for adding peripheral devices, such as game controllers, communications devices, and input devices, on a single bus.

The USB controller must be capable of waking the system as defined in Section 3.4.4 of the ACPI 1.0 specification. This capability is part of the requirement for ACPI compliance, as defined in the requirement, “System design meets ACPI 1.0 specification and PC 98 requirements,” earlier in this chapter.

The USB implementation in the system must also meet the requirements defined in the USB specifications plus any additional requirements for PC 98 defined in the “USB” chapter in Part 3 of this guide.

26. System includes support for other high-speed expansion capabilities

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Required, with 2 IEEE 1394 ports*

Recommended: Use either SCSI or IEEE 1394 as the default I/O port for devices such as scanners and external drives that require high-speed data transfer.

For all PC 98 systems, additional support for expansion capabilities can be provided using PCI 2.1, IEEE 1394, CardBus, or other high-speed buses. Any expansion bus implemented in the system must meet the requirements defined in the related chapter in Part 3 of this guide. For information about Device Bay requirements, see the “Device Bay-capable bay and peripherals meet Device Bay specification” requirement earlier in this chapter.

For Net PC systems, external expansion capabilities that are accessible by the end user are not recommended, with the exception of USB and Device Bay. If expansion capabilities are implemented in a Net PC system, additional requirements are defined in the *Network PC System Design Guidelines.*

27. If present, PCI bus meets PCI 2.1 or higher, plus PC 98 requirements

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

If PCI is used in a PC system, the PCI bus must meet the requirements defined in PCI 2.1 or higher, plus any additional requirements for PC 98 defined in the "PCI" chapter in Part 3 of this guide. Exceptions for particular devices are noted in Parts 3 and 4 of this guide.

Add-on PCI IDE devices must comply with PCI 2.1 requirements and also must provide Subsystem IDs and Subsystem Vendor IDs. However, PCI-to-PCI bridges and core chip sets do not have to provide Subsystem IDs and Subsystem Vendor IDs.

All PCI connectors on the system board must be able to allow any PCI expansion card to have bus master privileges.

28. System does not include ISA expansion devices

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required, with no ISA slots*

ISA expansion devices cannot be included in a PC 98 system, although ISA expansion slots are allowed on Consumer PC 98 and Office PC 98 systems. It is acceptable for all PC 98 systems to use ISA protocols and signaling or ISA-like protocols and signaling for an internal implementation. On-board devices such as audio, modems, or network adapters are not acceptable for PC 98, nor can the IDE controller use an ISA bus.

There will be a delayed phase-in period for this requirement, during which only system-board ISA devices will be allowed. Please notice that the ISA requirements that apply during the transition phase are the same as those defined in *PC 97 Hardware Design Guide,* and are also summarized in the "Legacy Support" appendix in the References part of this guide.

For Entertainment PC 98, the system must not contain ISA expansion slots that can be accessed by the end user. This is also a requirement for Net PC systems and is strongly recommended for workstations.

The benefits of designing ISA-free systems include easier and more stable system configuration, lower support cost, and improved performance.

For on-board legacy implementations, interrupts are supported using the legacy 8259 or (for Windows NT 5.0) APIC. Any on-board legacy implementations, such as BIOS ROM, Super I/O, 8042 controllers, math coprocessors, and so on, are allowed and must meet the requirements defined in the "Legacy Support" appendix in the References part of this guide.

Basic PC 98 I/O Devices

This section defines the general requirements for I/O devices. Additional requirements are defined in the “Basic PC 98 Graphics Adapters, Video, and Broadcast Services” and “Basic PC 98 Storage and Related Peripherals” sections later in this chapter.

29. System includes keyboard connection and keyboard

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required; USB or wireless*

Recommended: USB.

The external keyboard connection requirements on any PC can be met by using either USB, a PS/2-style port, or wireless capabilities in the system. A mobile or all-in-one system that has a built-in keyboard must also provide the capability for an external keyboard connection, which can be implemented using a port replicator or a single PS/2-style port with special cabling for both an external keyboard and an external mouse. For complete requirements for keyboard ports and peripherals, see the “I/O Ports and Devices” chapter in Part 4 of this guide.

30. System includes pointing-device connection and pointing device

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required; USB or wireless*

Recommended: USB or other external bus.

The external pointing-device connection requirements on any PC can be met by using USB, a PS/2-style port, or wireless capabilities in the system. A mobile or all-in-one system that has a built-in pointing device must also provide the capability for an external pointing-device connection. This can be implemented using a port replicator or a single PS/2-style port with special cabling for both an external keyboard and an external pointing device.

A second serial port is not an acceptable external connection for a pointing device. For complete requirements for pointing-device ports and peripherals, see the “I/O Ports and Devices” chapter in Part 4 of this guide.

31. System includes connection for external parallel devices

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

Recommended: USB or other external bus.

The requirement for an external connection for parallel devices can be met by using USB or another external bus. This capability can also be provided as a parallel port with extended capabilities port (ECP) capabilities, but a legacy parallel port is not the recommended implementation. For complete parallel port requirements, see the “I/O Ports and Devices” chapter in Part 4 of this guide.

32. System includes connection for external serial devices

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

Recommended: USB or CardBus.

The requirement for an external connection for serial devices can be met by using USB or CardBus. An RS-232C serial connection can also be implemented using a 16550A or equivalent serial port, but a legacy serial port is not recommended. For complete serial port requirements, see the “I/O Ports and Devices” chapter in Part 4 of this guide.

33. System includes wireless capabilities

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Recommended*

Infrared Data Association (IrDA) fast IR is recommended for all PC 98 systems, including mobile PCs, in order to enable synchronization and data exchanges with digital still cameras, printers, and other peripherals.

Standards for wireless PC peripherals are being developed within IrDA and are expected to be completed in 1997. For information about wireless requirements for PC 98, see the “I/O Ports and Devices” chapter in Part 4 of this guide.

34. System includes support for operating system installation

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

A PC 98 system must include I/O device support and system BIOS support for boot devices, as described in the “BIOS meets PC 98 requirements for boot support” requirement earlier in this chapter. This allows the user to install or reinstall the operating system. Legacy and native hard disks, CD-ROM, and so on can be connected using interfaces to IEEE 1394.

The PC 98 support requirement can be included as one of the following types of devices, depending on specific customer requirements:

- CD-ROM drive

  If this device is present, the host controller must meet the specific requirements defined in Part 3 of this guide, and the device must meet the requirements defined in the “Storage and Related Peripherals” chapter in Part 4.

- Network adapter

  For PC 98, the network adapter driver must support NDIS 5.0 to take advantage of new operating system capabilities. An NDIS 4.0 or MAC implementation is not acceptable. For complete information about PC 98 requirements for network adapters, see the “Network Communications” chapter in Part 4 of this guide.

  For a PC 98 system that uses a network adapter as a boot device to support operating system installation, the network adapter must be compatible with remote operating system installation and boot capabilities, as defined in the “Network Communications” chapter in Part 4 of this guide.

  For Office PC 98, the BIOS must support using a network adapter as a boot device, and the system must be tested with a network adapter as part of PC 98 compliance testing. For Net PC systems, a network adapter and system BIOS support are also required for using the adapter as a boot device.

- Floppy disk drive: 3.5-inch, 1.44-MB minimum capacity

  Support for a floppy disk drive can be provided using PC Card, USB, or a legacy FDC device as defined in the “Basic PC 98 Storage and Related Peripherals” section later in this chapter.

For the Windows NT operating system, either a CD-ROM drive or network adapter must be included in the system, because Windows NT does not support installing the operating system from a floppy disk drive.

Note: It is recognized that OEMs supply PC systems to corporations with specific feature requirements. For example, a customer might want to insert network adapters at the end-user site. However, a system submitted for PC 98 compliance testing must include a device that meets this requirement.

For information about how these requirements can be implemented for mobile PCs, see the “Mobile PC 98” chapter in Part 2 of this guide.

35. System audio support meets PC 98 requirements

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Required*

Although audio is a standard feature in most PC market segments, it is understood that certain SOHO and Office PC designs that focus on cost will not require audio. For those PCs that contain audio, the PC 98 audio requirements differ greatly from those defined in earlier hardware design guides, and they now include new performance metrics. For implementation details, see the “Audio Components” chapter in Part 4 of this guide.

36. System includes communications device

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required; modem	*Required; network adapter*	*Required; modem*

Recommended: NDIS 5.0-supported, high-speed dial-up communications.

The PC 98 minimum modem requirements call for an internal 33.6-Kbps V.34-1996 data/fax/voice modem, using WDM driver support where relevant. The increased PC 98 requirements represent current market trends for modems available in 1998.

For complete information about PC 98 requirements for communications devices, see the “Modems” and “Network Communications” chapters in Part 4 of this guide. For information about implementation of this requirement for mobile PCs, see the “Mobile PC 98” chapter in Part 2.

Note: It is recognized that OEMs supply PC systems to corporations with specific feature requirements. For example, a customer might want to insert network adapters at the end-user site. However, a Consumer PC 98 or Entertainment PC 98 system submitted for PC 98 compliance testing must include a modem that meets PC 98 requirements. An Office PC 98 system submitted for compliance testing must include a network adapter.

37. System includes smart card support

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Not applicable	*Recommended*	*Not applicable*

If implemented in an Office PC 98 system, smart card readers and cards must be compatible with *Interoperability Specification for ICCs and Personal Computer Systems*, published by Bull Personal Transaction Systems, Hewlett-Packard, Microsoft, Schlumberger, and Siemens Nixdorf at http://www.smartcardsys.com.

In addition, smart card readers and device drivers must be Plug and Play-compliant and must adhere to the Win32 smart card specifications published in the Windows NT DDK. Smart card applications and service-provider dynamic link libraries (DLLs) must adhere to the Win32 smart card specifications as published in the Win32 SDK.

The smart card system required for a PC 98 system with digital satellite television support represents a different technology, as defined in the “Video and Broadcast Components” chapter in Part 4 of this guide.

Basic PC 98 Graphics Adapters, Video, and Broadcast Services

This section summarizes the PC 98 requirements for graphics adapters and monitors. For complete details, including recommendations for hardware acceleration, see the “Graphics Adapters” chapter in Part 4 of this guide.

38. Graphics adapter meets PC 98 minimum requirements

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

The following list summarizes the key requirements for PC 98 graphics adapters:

- Graphics adapter uses PCI, AGP, or other high-speed bus.

  For the primary graphics adapter, the video bus must not use ISA or any other legacy local bus. Possible implementations that meet this requirement can include PCI 2.1 or the AGP 1.0 interface. For Entertainment PC 98, AGP is required.

  Whether implemented using PCI or AGP, the video bus must meet the requirements defined in the “Graphics Adapters” chapter in Part 4 of this guide.

- Graphics adapter works normally with default VGA mode driver.

  The default VGA driver is required for operating system installation. The adapter must support 4-bit planar VGA mode as described in the Windows DDK.

- Adapter and driver support multiple adapters and multiple monitors.

  This support ensures that the end user has guaranteed automatic support in the hardware and driver to allow the operating system to correctly configure use of multiple monitors or multiple graphics adapters.

- The adapter must support screen resolutions as defined by VESA up to the PC 98 required maximum, including 640 × 480 × [8, 15, 16, 24] bpp, 800 × 600 × [8, 15, 16, 24] bpp, and 1024 × 768 × [8, 15, 16] bpp.

For information about requirements for the built-in display adapter on a mobile PC, see the “Mobile PC 98” chapter in Part 2.

39. Adapter supports television output if system does not include a large-screen monitor

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Recommended*

The ability to connect and use a standard NTSC or PAL television as a large display surface is key to the ability to deliver realistic television, movie, and game experiences. For complete information about the television output requirements, refer to the "Graphics Adapters" chapter in Part 4 of this guide. For information about large-screen monitor requirements for Entertainment PC systems, see the "Monitors" chapter in Part 4 of this guide.

40. Color monitor is DDC-compliant with unique EDID identifier

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

A monitor designed for or included with a PC 98 system must be compliant with *Display Data Channel Standard, Version 2.0.* It also must transmit an Extended Display Identification Data (EDID) structure containing unique Manufacturer Name IDs and Product Code IDs, plus all required fields defined in Section 3 of the *Extended Display Identification Data Standard, Version 2.0* or higher. This requirement does not apply to liquid crystal displays (LCDs).

For complete PC 98 requirements for monitors, including requirements for image color matching (ICM), ergonomic timing standards, and DDC support, see the "Monitors" chapter in Part 4 of this guide.

41. System meets PC 98 DVD-Video and MPEG-2 playback requirements

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required with DVD-Video*	*Required*

Under Windows and Windows NT, operating-system playback support for MPEG-1 is provided through Microsoft DirectShow. This requirement refers to built-in system support for DVD-Video playback or any other MPEG-2 playback capabilities, whether provided as a hardware decoder, a software decoder, or a combination of the two. This requirement does not apply for Office PC 98 systems that provide DVD-ROM drives for storage purposes only.

For PC 98, the graphics adapter requirements in support of MPEG-2 and DVD-Video playback are described in the "Graphics Adapters" chapter in Part 4 of this guide. Software requirements and hardware capabilities for MPEG-2 and DVD-Video playback are defined in the "Video and Broadcast Components" chapter in Part 4 of this guide.

These capabilities are recommended for mobile PCs, as described in the "Mobile PC 98" chapter in Part 2 of this guide.

42. System supports PC 98 analog video input and capture capabilities

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Required*

If video-capture capability is implemented in a PC 98 system, it must meet the requirements defined in the “Video and Broadcast Components” chapter in Part 4 of this guide.

Support for video input and capture is recommended for Consumer PC 98and Office PC 98 systems, implemented as an add-on device or a direct interface on the system board. Systems with USB or IEEE 1394 support are capable of supporting the new low-cost digital video cameras entering the Office PC 98 market. It is strongly recommended that systems include more than one USB or IEEE 1394 port if the PC comes bundled with a USB or IEEE 1394 video conferencing camera.

For PC 98, all video input sources and capture devices must implement driver support as defined for Still Image architecture in the Windows NT 5.0 DDK.

43. System includes analog television tuner

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Required*

Recommended: Digital broadcast or satellite television tuner.

If this capability is implemented in a PC 98 system, it must meet the requirements defined in the “Video and Broadcast Components” chapter in Part 4 of this guide.

Basic PC 98 Storage and Related Peripherals

This section summarizes the PC 98 requirements for storage devices. For system requirements related to CD-ROM and floppy disk drives, see the “Basic PC 98 Buses and Devices” section earlier in this chapter.

44. System BIOS and option ROMs support Int 13h Extensions

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

This requirement applies for systems that run either Windows or Windows NT, but does not apply for RISC-based PCs.

The Int 13h Extensions ensure correct support for high-capacity drives. Support for the fixed-disk access subset of Int 13h Extensions must be provided in the system BIOS and in any option ROMs for storage devices that include BIOS support. The Int 13h Extensions are defined in the Windows NT 5.0 DDK and in the “Layered Block Device Drivers” section of the Windows DDK.

45. Host controller for storage device meets PC 98 requirements

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

The host controller in a PC 98 system must meet requirements defined for the bus it uses. For IDE or SCSI controllers, it must also meet the requirements outlined in the "IDE and ATAPI" or "SCSI" chapters in Part 3 of this guide.

46. Host controllers and devices support bus mastering

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

The host controller for storage devices must support bus mastering, whether using IDE, SCSI, or IEEE 1394. Bus mastering support must also be enabled for storage devices, including hard disks, CD-ROM, DVD-ROM, and tape drives. Use of the ISA bus by storage devices is not acceptable for PC 98 systems.

Bus master capabilities must meet the related specification for the particular controller. For example, the programming register set for PCI IDE bus master DMA is defined in Small Form Factor (SFF) 8038i.

Note: This requirement does not apply to legacy FDCs and will not become a requirement for FDCs in the future.

47. Hard drive meets PC 98 requirements

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

The hard disk drive must meet the requirements defined in the "Storage and Related Peripherals" chapter in Part 4 of this guide. Hard drives implemented as IDE or SCSI peripherals must also meet the requirements outlined in the "IDE and ATAPI" or "SCSI" chapters in Part 3 of this guide.

For all PC 98 systems, the hard drive must be SMART-compliant, using the SMART I/O control application programming interface (IOCTL API), as defined in the "Storage and Related Peripherals" chapter in Part 4 of this guide.

48. Removable media support media status notification

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

CD-ROM and DVD-ROM devices must comply with all provisions defined in the "Media Status Event Notification Support" subsection of SFF 8090 (Mt. Fuji specification). Other types of IDE and ATAPI removable devices must follow the *Media Status Notification Support Specification, Version 1.03.*

For SCSI devices other than CD-ROM and DVD, additional media status notification support is not required.

49. Floppy disk capabilities are provided using an expansion card or external bus

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Recommended*

To support migration away from legacy devices for PC 98, it is recommended that support for floppy disk drives be provided by using a solution based on an external bus, such as USB, PC Card, or a SCSI or IDE expansion card. If a legacy FDC is included on a PC 98 system, it must meet the requirements specified in the “Storage and Related Peripherals” chapter in Part 4 of this guide.

For Net PC systems, any floppy disk drive implemented on the system must be capable of being remotely disabled and locked as defined in the *Network PC System Design Guidelines.* This is also recommended for Office PC 98.

Manageability Component Instrumentation Requirements

This section presents new requirements and recommendations for PC 98 systems related to the Wired for Management (WfM) initiative and the Zero Administration initiative for Windows. The WfM initiative seeks to raise the level of management capabilities for mobile, desktop, and server platforms. The Zero Administration initiative seeks to ensure a controlled, highly manageable enterprise.

The baseline for these requirements is *Windows Hardware Instrumentation Implementation Guidelines, Version 1.0* (WHIIG), which also defines the Windows-specific requirements of the *Wired for Management Baseline Specification, Version 2.0,* for hardware instrumentation.

Collectively, the items in this section represent the Manageability Baseline requirements for Office PC 98. Platform management information requirements are defined for two key areas:

- Component instrumentation. Interfaces through which information is supplied by PC 98 platform management components.
- Management information providers. Interfaces used by applications to access PC 98 platform management information.

Tips for implementing management capabilities. For PC 98 systems and components, these are the design steps to pursue:

- Implement the component instrumentation features defined for PC 98 systems in WHIIG.
- For those components that require Windows Management Instrumentation (WMI), ensure that WMI is enabled in device minidrivers as defined in the Windows NT 5.0 DDK.
- Refer to WHIIG for other driver requirements and design tips.
- For all instrumented components, test against the baseline features required in WHIIG.
- For each component, extend the Web-Based Enterprise Management (WBEM) and Common Information Model (CIM) schema to expose the device's custom features in any CIM-ready management browser.

50. System supports WHIIG

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Not applicable	*Required*	*Not applicable*

The PC 98 requirement is defined in *Windows Hardware Instrumentation Implementation Guidelines, Version 1.0.*

51. System includes driver support for WMI

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Not applicable	*Required*	*Not applicable*

Requirements and recommendations related to implementing WMI for Windows NT 5.0 and Windows are defined in WHIIG.

Support for WMI, CIM, and Win32 extension schema objects and data must be implemented as defined in WHIIG.

52. Management information service provider enabled by default

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Not applicable	*Required*	*Not applicable*

The management information service providers must be enabled on Office PC 98 and Net PC systems as defined in WHIIG.

Also, newly developed applications for managing WBEM-capable PC 98 systems should be written to access those systems through industry-standard WBEM protocols and interfaces.

53. Expansion devices can be remotely managed

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Not applicable	*Required*	*Not applicable*

Devices provided as expansion devices must be capable of being remotely managed to ensure that control and TCO policies can be realized. For example, for any implementation of a floppy disk drive on an Office PC 98 system, the drive must be capable of being remotely disabled as a boot selection and provisions must be made for locking.

It is not a requirement that certain devices be capable of being remotely disabled, including the primary hard disk drive, the network adapter, and any standard devices that use legacy connections, such as a keyboard or pointing device that uses a PS/2 connection. However, it must be possible that permissions, policies, or other methods can be used to remotely manage capabilities such as hard disk access or to control end-user ability to change the MAC address or configuration settings for the network adapter.

See also the "BIOS meets PC 98 requirements for boot support" requirement in the "Basic PC 98 General System Requirements" section earlier in this chapter.

Basic PC 98 References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

Advanced Configuration and Power Interface Specification, Revision 1.0
http://www.teleport.com/~acpi/

Device Bay Interface Specification, Version 1.0
http://www.device-bay.org

Display Data Channel Standard, Version 2.0
Extended Display Identification Data Standard, Version 2.0
http://www.vesa.org

El Torito—Bootable CD-ROM Format Specification, Version 1.0
Compaq, Intel, Phoenix BIOS Boot Specification, Version 1.01
http://www.ptltd.com/techs/specs.html

Hot-plugging support under Windows operating systems
http://www.microsoft.com/hwdev/devdes/

Intel/Duracell Smart Battery System Specification
http://www.mediacity.com/~sbs/

Intel hardware developer site
http://developer.intel.com

Interoperability Specification for ICCs and Personal Computer Systems
http://www.smartcardsys.com

Media Status Notification Support Specification, Version 1.03
Plug and Play specifications
http://www.microsoft.com/hwdev/specs/
Vendor ID registration: pnpid@microsoft.com

MultiProcessor Specification, Version 1.4
Intel part number 242016-002
http://developer.intel.com

PCI Local Bus Specification, Revision 2.1 (PCI 2.1)
PCI SIG
Phone: (800) 433-5177
http://www.pcisig.com

Power management specifications for device and bus classes
Guidelines for audible noise and other OnNow technologies
http://www.microsoft.com/hwdev/onnow.htm

SFF 8070i, SFF 8038i, SFF 8090 (Mt. Fuji specification), and other SFF specifications
SFF Committee publications
FaxAccess: (408) 741-1600 (fax-back)
Fax: (408) 867-2115
ftp://ftp.symbios.com/pub/standards/io/

Universal Serial Bus PC Legacy Compatibility Specification, Version 0.9
http://www.teleport.com/~usb/data/usb_le9.pdf

USB Specification, Version 1.0
USB Device Class Definition for Human Interface Devices, Version 1.0
Other USB device class specifications
Phone: (503) 264-0590
Fax: (503) 693-7975
http://www.usb.org

Web-Based Enterprise Management (WBEM) information
http://wbem.freerange.com
http://www.microsoft.com/management/wbem/
http:www.dmtf.org/work/cim.html

Windows and Windows NT DDKs, including NDIS documentation
MSDN Professional membership

Windows Hardware Instrumentation Implementation Guidelines, Version 1.0 (WHIIG), Microsoft Corporation and Intel Corporation
http://www.microsoft.com/hwdev/specs/
(This specification is expected in the second half of 1997.)

Wired for Management Baseline Specification, Version 2.0
Intel Corporation.
http://www.intel.com/managedpc/wired
(This specification is expected in the second half of 1997.)

Checklist for Basic PC 98

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

Consumer PC 98	Office PC 98	Entertainment PC 98
1. System performance meets PC 98 minimum requirements		
Required	*Required*	*Required*
2. System design meets ACPI 1.0 specification and PC 98 requirements		
Required	*Required*	*Required*
3. Hardware design supports OnNow initiative		
Required	*Required*	*Required*
4. BIOS meets PC 98 requirements for OnNow support		
Required	*Required*	*Required*
5. BIOS meets PC 98 requirements for boot support		
Required	*Required*	*Required*
6. All expansion slots in the system are accessible for users to insert cards		
Required	*Required*	*Required*
7. Audible noise meets PC 98 requirements		
Required	*Required*	*Required*
8. System and component design practices follow accessibility guidelines		
Recommended	*Recommended*	*Recommended*
9. Internal system modification capabilities are not accessible to end users		
Recommended	*Recommended*	*Recommended*
10. System design provides physical security		
Recommended	*Recommended*	*Recommended*
11. Each device and driver meets PC 98 device requirements		
Required	*Required*	*Required*
12. Each bus and device meets Plug and Play specifications		
Required	*Required*	*Required*
13. Unique Plug and Play device ID provided for each system device and add-on device		
Required	*Required*	*Required*
14. Option ROMs meet Plug and Play requirements		
Required	*Required*	*Required*
15. "PNP" vendor code used only to define a legacy device's CompatibleID		
Required	*Required*	*Required*
16. Device driver and installation meet PC 98 requirements		
Required	*Required*	*Required*
17. Minimal user interaction needed to install and configure devices		
Required	*Required*	*Required*
18. Connections use icons plus keyed or shrouded connectors		
Required	*Required*	*Required*
19. Hot-plugging capabilities for buses and devices meet PC 98 requirements		
Required	*Required*	*Required*

Consumer PC 98	Office PC 98	Entertainment PC 98
20. Device Bay-capable bay and peripherals meet Device Bay specification		
Required	*Required*	*Required*
21. Multifunction add-on devices meet PC 98 device requirements for each device		
Required	*Required*	*Required*
22. All devices support correct 16-bit decoding for I/O port addresses		
Required	*Required*	*Required*
23. System-board devices use ISA-compatible addresses		
Required	*Required*	*Required*
24. Each bus meets written specifications and PC 98 requirements		
Required	*Required*	*Required*
25. System includes USB with one USB port, minimum		
Required	*Required*	*Required, with 2 USB ports*
26. System includes support for other high-speed expansion capabilities		
Recommended	*Recommended*	*Required, with 2 IEEE 1394 ports*
27. If present, PCI bus meets PCI 2.1 or higher, plus PC 98 requirements		
Required	*Required*	*Required*
28. System does not include ISA expansion devices		
Required	*Required*	*Required, with no ISA slots*
29. System includes keyboard connection and keyboard		
Required	*Required*	*Required; USB or wireless*
30. System includes pointing-device connection and pointing device		
Required	*Required*	*Required; USB or wireless*
31. System includes connection for external parallel devices		
Required	*Required*	*Required*
32. System includes connection for external serial devices		
Required	*Required*	*Required*
33. System includes wireless capabilities		
Recommended	*Recommended*	*Recommended*
34. System includes support for operating system installation		
Required	*Required*	*Required*
35. System audio support meets PC 98 requirements		
Recommended	*Recommended*	*Required*
36. System includes communications device		
Required; modem	*Required; network adapter*	*Required; modem*
37. System includes smart card support		
Not applicable	*Recommended*	*Not applicable*
38. Graphics adapter meets PC 98 minimum requirements		
Required	*Required*	*Required*
39. Adapter supports television output if system does not include a large-screen monitor		
Recommended	*Recommended*	*Recommended*
40. Color monitor is DDC-compliant with unique EDID identifier		
Required	*Required*	*Required*

Consumer PC 98	Office PC 98	Entertainment PC 98
41. System meets PC 98 DVD-Video and MPEG-2 playback requirements		
Required	*Required with DVD-Video*	*Required*
42. System supports PC 98 analog video input and capture capabilities		
Recommended	*Recommended*	*Required*
43. System includes analog television tuner		
Recommended	*Recommended*	*Required*
44. System BIOS and option ROMs support Int 13h Extensions		
Required	*Required*	*Required*
45. Host controller for storage device meets PC 98 requirements		
Required	*Required*	*Required*
46. Host controllers and devices support bus mastering		
Required	*Required*	*Required*
47. Hard drive meets PC 98 requirements		
Required	*Required*	*Required*
48. Removable media support media status notification		
Required	*Required*	*Required*
49. Floppy disk capabilities are provided using an expansion card or external bus		
Recommended	*Recommended*	*Recommended*
50. System supports WHIIG		
Not applicable	*Required*	*Not applicable*
51. System includes driver support for WMI		
Not applicable	*Required*	*Not applicable*
52. Management information service provider enabled by default		
Not applicable	*Required*	*Not applicable*
53. Expansion devices can be remotely managed		
Not applicable	*Required*	*Not applicable*

C H A P T E R 4

Workstation PC 98

This chapter provides a summary of the key requirements for workstations designed as PC 98 systems. If there is a conflict with requirements or recommendations made elsewhere in this guide, the items in this chapter have precedence for workstations. Unless a specific requirement or exception is defined in this chapter, the requirements defined in the "Basic PC 98" chapter apply for workstations.

Important: The system requirements defined in this chapter provide guidelines for designing PC systems that will result in the optimal user experience with typical Windows-based applications running under the Microsoft Windows NT Workstation operating system only. These design requirements are not basic system requirements for running the Windows NT operating system.

Contents

Workstation Platform Guidelines

The goal of this chapter is to describe requirements that define a workstation that is optimized to run Windows NT Workstation and to support Win32-based applications created for the workstation. The workstation is a platform for users whose principal computing tasks involve running mission-critical networked applications, engineering or scientific applications, media-authoring tools, or software-development tools.

Although Windows NT Workstation is used on stand-alone systems, the PC 98 requirements support the more common use of Windows NT Workstation as a platform for network productivity.

The key design issues for PC 98 workstations include processor, memory, and bus architecture requirements to support intensive computational activities.

1. Workstation meets all requirements for Office PC 98
Required
Each Basic PC 98 component indicated as a requirement for Office PC 98 is also a requirement for PC 98 workstations.

Note: Systems designed to run only on Windows NT are not required to meet PC 98 requirements for legacy Plug and Play support. If the system is designed to run either Windows 98 or Windows NT, it must meet all PC 98 requirements for legacy Plug and Play support.

2. Workstation meets requirements for optimal performance
Required
In addition to the basic PC 98 minimum requirements, the following items are required for optimal workstation performance:

- Pentium Pro or Pentium II processor, RISC-based processor, or equivalent performance.
- 64-MB RAM.

  For multiprocessor systems, 64 MB minimum is required for each processor.
- Minimum 512K L2 cache (per processor for multiple processors).

3. Workstation supports multiple processors
Recommended
Multiprocessor support using Intel Architecture processors must comply with *MultiProcessor Specification, Version 1.4* or higher, from Intel Corporation.

If multiprocessor support is provided, each processor must have a separate L2 cache.

4. Workstation RAM can be expanded

Recommended

Workstation RAM should be capable of being expanded to at least 1 GB.

5. Workstation system memory includes ECC memory protection

Required

The system memory and L2 cache must be protected with Error Correction Code (ECC) memory protection. The ECC must be able to detect a double-bit error in one word and to correct a single-bit error in one word, where "word" means the base width in bits of the memory subsystem. To detect the failure of a single DRAM device, the ECC should be capable of detecting a 4-bit or 8-bit error in one word, with detection of 4-bit errors preferred. An error that cannot be corrected must result in a system fault.

6. Workstation includes APIC support

Required

The workstation must include APIC support, implemented as defined in the APIC extension to the ACPI 1.0 specification. Features such as targeted interrupts, broadcast interrupts, and prior-owner interrupts must be supported. Intel Architecture processor implementations can use the Intel APIC component.

7. Workstation includes high-performance components

Recommended

Basic PC 98 requirements support high-performance components for workstations, such as bus mastering for I/O and storage and write combining for Pentium Pro and Pentium II processors. In addition, the PC 98 recommendation for ensuring that drivers are tuned for 32-bit performance is especially recommended for workstations.

8. Workstation supports 64-bit I/O bus architecture

Recommended

For PCI, the workstation should support the 64-bit physical address space, and PCI adapters should be able to address any location in that address space.

9. Workstation does not include ISA expansion slots

Recommended

It is strongly recommended that workstations not include ISA expansion slots. ISA devices cannot meet the high-performance requirements for workstation systems, resulting in a performance bottleneck.

10. Graphics subsystem supports workstation performance demands
Required

This requirement is for workstations designed to support high-resolution graphics applications. A workstation does not have to meet this requirement if it is designed for financial or transaction-based markets and is not intended to support graphics-intensive applications.

For a workstation PC that is intended to support graphics-intensive applications the following PC 98 support must be provided:

- For Pentium Pro and Pentium II or compatible processors, 4 MB of display RAM and support for write-combining optimizations under Windows NT.
- Support for 3-D hardware acceleration with DirectX support or OpenGL acceleration.

  Direct3D hardware designed to support OpenGL-based applications must be capable of meeting the OpenGL rasterization rules. Direct3D drivers must report through the appropriate capabilities bit whether or not the hardware actually conforms to OpenGL requirements.

  For information about 3-D hardware acceleration supported by Direct3D, see the "Graphics Adapters" chapter in Part 4 of this guide. For information about OpenGL rasterization requirements and conformance rules, see the web site at http://www.sgi.com/technology/opengl/arb.html.

For workstation systems intended for use with computer-aided design (CAD) or other high-performance graphical applications, 1280 × 1024 × 24 bpp resolution is recommended. Hardware that implements 32-bpp display modes (for example, display hardware for high-end engineering workstations) should implement RGB-mode rasterization.

11. Workstation meets PC 98 DVD-Video and MPEG-2 playback requirements
Required, with DVD-Video

Support for DVD-Video playback is required for a workstation only if the workstation supports playback on DVD-Video devices.

If the workstation does include a DVD-Video device, then the system must support MPEG-2 and DVD-Video playback as described in the "Graphics Adapters" and "Video and Broadcast Components" chapters in Part 4 of this guide.

12. Storage components rely on SCSI controller

Recommended

SCSI is a flexible I/O bus that supports good performance for access and throughput to meet a workstation's intensive data transfer needs. For more information about related requirements, see the "SCSI" chapter in Part 3 of this guide.

13. Workstation includes multiple hard drives

Recommended

Recommended: Hardware acceleration of RAID (redundant array of inexpensive disks) drives.

Multiple hard drives can be incorporated for improved performance (multiple spindle access and striping with RAID 0) or for data integrity (RAID 1/5).

Workstation PC 98 References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

Advanced Configuration and Power Interface Specification, Revision 1.0
http://www.teleport.com/~acpi/

Intel hardware developer site
http://developer.intel.com

MultiProcessor Specification, Version 1.4
Intel part number: 242016-002
http://developer.intel.com

OpenGL conformance rules from the OpenGL Architectural Review Board
http://www.sgi.com/technology/opengl/arb.html
UseNet news group for OpenGL at comp.graphics.opengl

Windows NT DDK
MSDN Professional membership

Checklist for Workstation PC 98

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

1. Workstation meets all requirements for Office PC 98
Required

2. Workstation meets requirements for optimal performance
Required

3. Workstation supports multiple processors
Recommended

4. Workstation RAM can be expanded
Recommended

5 Workstation system memory includes ECC memory protection
Required

6. Workstation includes APIC support
Required

7. Workstation includes high-performance components
Recommended

8. Workstation supports 64-bit I/O bus architecture
Recommended

9. Workstation does not include ISA expansion slots
Recommended

10. Graphics subsystem supports workstation performance demands
Required

11. Workstation meets PC 98 DVD-Video and MPEG-2 playback requirements
Required, with DVD-Video

12. Storage components rely on SCSI controller
Recommended

13. Workstation includes multiple hard drives
Recommended

C H A P T E R 5

Entertainment PC 98

5

This chapter provides a summary of the key requirements for Entertainment PC 98.

If there is a conflict with requirements or recommendations made elsewhere in this guide, the items in this chapter have precedence for Entertainment PC 98. Unless a specific requirement or exception is defined in this chapter, the requirements defined in the “Basic PC 98” chapter apply for Entertainment PC 98.

Important: The system requirements defined in this guide provide guidelines for designing PC systems that will result in the optimal user experience with typical Windows-based applications running under either the Microsoft Windows or Windows NT Workstation operating systems. These design requirements are not the basic system requirements for running the Windows operating system.

Contents

Entertainment PC 98 System Requirements

This section summarizes the requirements for the basic components of Entertainment PC 98 systems. System requirements for computational capabilities and memory are the same for Entertainment PC 98 as for all PC 98 systems.

1. Entertainment PC 98 system includes two USB ports, with at least one easily accessible connector
Required

The Entertainment PC 98 system must include two USB ports, with at least one USB connector in an easily accessible location (not on the rear panel). This is to support connection of auxiliary input devices such as game pads, joysticks, and track balls.

The recommended location for an easily accessible connector is to place a port on the front of the PC. The second connector for static connections to continuously used devices can be placed on the rear of the PC.

The USB implementation must meet the requirements defined in the “USB” chapter in Part 3 of this guide.

2. Entertainment PC 98 system includes two IEEE 1394 ports, with at least one easily accessible connector
Required

The Entertainment PC 98 system must include two IEEE 1394 ports, with at least one IEEE 1394 connector in an easily accessible location (not on the rear panel) to support camcorders and other digital consumer-electronics devices.

The recommended location for an easily accessible connector is to place a port on the front of the PC. The second connector for static connections to continuously used devices can be placed on the rear of the PC.

The IEEE 1394 implementation must meet the requirements defined in the “IEEE 1394” chapter in Part 3 of this guide.

3. Entertainment PC 98 system does not include ISA expansion slots
Required

For Entertainment PC 98, the system must not contain ISA expansion slots that can be accessed by the end user. The benefits of designing ISA-free systems include easier and more stable system configuration, lower support cost, and improved performance.

Any on-board legacy implementations, such as BIOS ROM, Super I/O, 8042 controllers, math coprocessors, and so on, must meet the requirements defined in the “Legacy Support” appendix in the References part of this guide.

4. All Entertainment PC 98 input devices meet USB HID specifications

Required

Recommended: Wireless connection for all input devices.

For Entertainment PC 98, all keyboards, pointing devices, and game pads and their connections must be compliant with the *USB Device Class Definition for Human Interface Devices, Version 1.0* or higher, and with the *USB HID Usages Table*, whether implemented as wired or wireless devices.

The game device must support at least four devices simultaneously, and its driver must support Microsoft DirectInput.

Note: For family-room PC design, a wireless connection is strongly recommended for the keyboard, pointing device, and game pad. For a pointing device, this requirement can be satisfied by using a remote-control pointing device or by using a wireless keyboard that includes a connector to enable a standard mouse to be attached. The ability to attach a standard two-button mouse is strongly encouraged, although the device itself does not need to be included with the PC.

For more information about input devices, see the “I/O Ports and Devices” chapter in Part 4 of this guide.

5. Entertainment PC 98 includes remote-control pointing device

Recommended

If a remote-control pointing device is implemented on an Entertainment PC 98 system, its command structure must be compliant with the *USB HID Usages Table*.

There is no requirement for implementing an IR or RF remote-control pointing device.

If included with an Entertainment PC 98 system, the remote-control device must have the following controls:

Label	Icon	Function
Power	—	Toggle switch between On and Standby power state
Start	Windows flag (under Microsoft licensing agreement)	Display Start menu (same action as the keyboard Windows logo key)
—	—	Mouse pointer control, including left and right mouse buttons

In addition, the following button is recommended for remote-control devices.

Label	Icon	Function
Cancel	—	Same as the keyboard ESCAPE key
Enter	—	Same as the keyboard ENTER key
Menu	—	Display application menus or toolbar (same as the keyboard F10 key)

All buttons and any additional controls implemented by the independent hardware vendor (IHV) must comply with the current USB HID specifications, including HID usage-code specifications as defined in the *USB HID Usages Table*.

Note: When the industry adopts standards for wireless controls, these will be incorporated as Entertainment PC system requirements.

6. All Entertainment PC 98 pointing devices support DirectInput and work simultaneously
Required

All pointing devices implemented in the Entertainment PC 98 system must have drivers that support Microsoft DirectInput. All pointing devices must also be able to correctly provide simultaneous input. This means that the pointing devices must not use competing protocols and that no pointing device is automatically disabled when another pointing device is in use. This also applies to game input devices in order to enable multiplayer gaming.

For information about implementing drivers that support simultaneous use of devices, see the Microsoft DirectX DDK.

7. Entertainment PC 98 includes data/fax/voice modem that supports V.pcm
Required

Recommended: NDIS 5.0-supported dial-up high-speed communications.

The Entertainment PC 98 minimum requirement for modem support is an internal PCM data/fax/voice modem, with WDM support where relevant. If the modem does not support V.pcm, it must be capable of being upgraded by end users to support V.pcm. For more information, see the “Modems” chapter in Part 4 of this guide.

8. Entertainment PC 98 includes DVD-ROM drive and DVD-Video playback
Required

The DVD-ROM device included in an Entertainment PC 98 system must meet the requirements defined for DVD in the “Storage and Related Peripherals” chapter in Part 4 of this guide. An Entertainment PC 98 system must also provide playback support for DVD-Video content, as defined in the “Video and Broadcast Components” chapter in Part 4 of this guide.

Entertainment PC 98 Audio Requirements

High-quality audio is a key differentiating feature for Entertainment PC 98. Audio fidelity and functions must be significantly better than for traditional PCs and on par with consumer-electronics stereos.

Using IEEE 1394 for positional 3-D audio and connections to home-theater systems will enable more-realistic game and video experiences. One opportunity for achieving this is to implement the audio subsystem as an external digital-to-analog converter (DAC) attached to a secondary IEEE 1394 port on the rear of the PC. This isolates the analog audio stream from the RF noise of internal PC components while enabling easy connection to either legacy analog or new Plug and Play-compatible digital stereo components.

9. Audio support meets PC 98 audio and Entertainment PC 98 requirements
Required
Recommended: Audio hardware accelerator is ready for digital audio.

Entertainment PC 98 audio must meet PC 98 audio requirements, which include requirements for audio hardware capabilities, performance metrics, and external connections. The following list summarizes the additional audio features required for Entertainment PC 98:

- Audio solution includes support for 3-D audio effects
- Input and output includes support for independent sample rates
- Audio solution supports music synthesis and echo cancellation

For more information, see the “Audio Components” chapter in Part 4 of this guide.

Entertainment PC 98 Graphics Components

This section summarizes the Entertainment PC 98 requirements for graphics adapters and monitors. For complete information about the requirements summarized in this section, see the “Graphics Adapters” chapter in Part 4 of this guide.

10. Graphics adapter uses AGP
Required
The graphics subsystem on an Entertainment PC 98 system must not use PCI, ISA, or VESA local bus (VLB). For information about implementing AGP, see the “Graphics Adapters” chapter in Part 4 of this guide.

Note: Integrated graphics subsystems that do not use AGP but meet or exceed AGP performance levels are acceptable for Entertainment PC 98.

11. Entertainment PC 98 graphics subsystem includes PC 98 hardware acceleration for 2-D and 3-D graphics

Required

Windows and Windows NT operating systems provide application programming interfaces (APIs) that accelerate graphics display through direct manipulation of video display memory, hardware blters, hardware overlays, and page flipping. Hardware-acceleration features for 2-D and 3-D graphics must be implemented to improve overall graphics performance, as defined in the "Graphics Adapters" chapter in Part 4 of this guide.

12. Entertainment PC 98 graphics subsystem includes support for television output if the system doesn't have a large-screen monitor

Recommended

Support for NTSC, PAL, or both types of television output is recommended, except for systems bundled with a large-screen super VGA (SVGA) monitor.

For Entertainment PC 98, the ability to connect to a television is key to its ability to deliver more realistic television, movie, and game experiences, and to enable social computing activities. Television output integrated with the PC graphics adapter will deliver much higher image quality than external converters. As such, this feature optimizes the usability of an Entertainment PC system connected directly to a television in the family room and for desktop systems configured to transmit graphics and video to a television in another room.

This capability must meet the PC 98 requirements for television output for composite and S-Video connectors, parameter control, and hardware filtering and scaling capabilities as defined in the "Graphics Adapters" chapter in Part 4 of this guide.

13. Entertainment PC 98 includes large-screen DDC2B color entertainment monitor

Recommended

A large-screen SVGA monitor that meets the *Display Data Channel Standard, Version 2.0*, Level B specification (DDC2B) is recommended for Entertainment PC 98 systems designed for the family room. Games, movies, and other entertainment software experiences are greatly enhanced by display screens comparable to modern television sizes—for example, 27 inches and larger in the United States.

An Entertainment PC 98 system that includes a large-screen monitor must meet the requirements for entertainment monitors defined in the "Monitors" chapter in Part 4 of this guide.

In addition, in 1999, large-screen entertainment monitors will likely be required to comply with interconnectivity standards currently in development as part of the VESA PC Theater initiative.

Entertainment PC 98 Video and Broadcast Components

This section summarizes the Entertainment PC 98 hardware requirements for video capture, television output, and DVD playback support. For complete information about the requirements summarized in this section, see the “Video and Broadcast Components” chapter in Part 4 of this guide.

14. Entertainment PC 98 DVD-Video and MPEG-2 playback meet PC 98 requirements
Required

DVD-Video support is required for Entertainment PC 98, so DVD playback capabilities must meet PC 98 requirements. If MPEG-2 hardware is included in an Entertainment PC 98 system, it must also meet PC 98 requirements.

15. Entertainment PC 98 supports PC 98 analog video input and capture capabilities
Required

Recommended: Additional support for digital video input using IEEE 1394.

WDM support must be implemented for all video input and capture capabilities as part of the PC 98 requirements. The video input connector should be easily accessible on the Entertainment PC 98 system (that is, not on the rear panel)

16. Entertainment PC 98 includes analog television tuner
Required

Recommended: Digital television tuner.

The NTSC or PAL decode component of the television tuner and analog video input subsystems must properly support extraction of data transmitted during the vertical blanking interval (VBI). This includes allowing certain scan lines to be placed within a separate memory buffer.

Note: The requirement for a cable-television tuner does not apply for PCs sold in Japan or other locales where cable television is not a common standard.

17. Entertainment PC 98 includes digital broadcast satellite subsystem
Recommended

If this capability is included in Entertainment PC 98, the implementation must include a digital broadcast satellite network card, a smart card, and drivers that meet PC 98 requirements.

18. Entertainment PC 98 includes DTV support
Required

Support for digital television (DTV) is required for Entertainment PC 98 starting in 1999, the expected time frame for initial digital television broadcasts. This includes hardware and software support for an ATSC tuner/demodulator, MPEG-2 decode capabilities, and graphics adapter support, as defined in the "Video and Broadcast Components" chapter in Part 4 of this guide.

As with all PC 98 components, compliance testing will begin when all related components are generally available.

Entertainment PC 98 References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

Display Data Channel Standard, Version 2.0 Level B (DDC2B)
Video Electronics Standards Association (VESA)
http://www.vesa.org

Windows and Windows NT DDKs
MSDN Professional membership

USB Device Class Definition for Human Interface Devices, Version 1.0
USB HID Usages Table
USB Implementers Forum
http://www.usb.org

Checklist for Entertainment PC 98

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

1. Entertainment PC 98 system includes two USB ports, with at least one easily accessible connector
Required

2. Entertainment PC 98 system includes two IEEE 1394 ports, with at least one easily accessible connector
Required

3. Entertainment PC 98 system does not include ISA expansion slots
Required

4. All Entertainment PC 98 input devices meet USB HID specifications
Required

5. Entertainment PC 98 includes remote-control pointing device
Recommended

6. All Entertainment PC 98 pointing devices support DirectInput and work simultaneously
Required

7. Entertainment PC 98 includes data/fax/voice modem that supports V.pcm
Required

8. Entertainment PC 98 includes DVD-ROM drive and DVD-Video playback
Required

9. Audio support meets PC 98 audio and Entertainment PC 98 requirements
Required

10. Graphics adapter uses AGP
Required

11. Entertainment PC 98 graphics subsystem includes PC 98 hardware acceleration for 2-D and 3-D graphics
Required

12. Entertainment PC 98 graphics subsystem includes support for television output if the system doesn’t have a large-screen monitor
Recommended

13. Entertainment PC 98 includes large-screen DDC2B color entertainment monitor
Recommended

14. Entertainment PC 98 DVD-Video and MPEG-2 playback meet PC 98 requirements
Required

15. Entertainment PC 98 supports PC 98 analog video input and capture capabilities
Required

16. Entertainment PC 98 includes analog television tuner
Required

17. Entertainment PC 98 includes digital broadcast satellite subsystem
Recommended

18. Entertainment PC 98 includes DTV support
Required

C H A P T E R 6

Mobile PC 98

This chapter provides a summary of the key PC 98 requirements for mobile PCs, mini-notebooks, docking stations, and port replicators. If there is a conflict with requirements or recommendations made elsewhere in this guide, the items in this chapter have precedence for mobile PCs. Unless a specific requirement or exception is defined in this chapter, the requirements defined in the “Basic PC 98” chapter apply for mobile PCs.

Important: The system requirements defined in this guide provide guidelines for designing PC systems that will result in the optimal user experience with typical Windows-based applications running under either the Microsoft Windows or Windows NT Workstation operating systems. These design requirements are *not* the basic system requirements for running the Windows operating system.

Contents

Introduction to Mobile PC Platform Guidelines

Many of the features required for Basic PC 98 demand significant power and heat dissipation, which will not work physically in the notebook environment. For mobile PC users, the issues of greater importance are portability (weight) and availability (battery life). The mini-notebook, an emerging class of machines, presents new demands with more stringent constraints that also must be addressed.

New external buses, support for multimedia applications, and other changes in PCs challenge mobile PC designers to incorporate the features that users want in a way that does not reduce the value of the core system.

The overall goal for mobile PC design is the same as for PC 98 design—enhanced user experience—but the design tradeoffs are different. This section presents mobile PC requirements in a manner that will encourage industry innovation across a wide range of design solutions without creating extreme power demands. Mobile PC requirements allow OEMs the flexibility they need to manage power and heat considerations in their designs.

Because OnNow and ACPI standards are now spreading throughout the industry, 1998 represents a transition for mobile computing. The new standards allow the operating system to take over the critical operations, such as dynamic configuration and power management, in support of mobile PCs.

The key mobile PC design issues include:

- Low weight and small size.
- Available battery life to meet user expectations.
- Power demands and heat dissipation on notebook computers managed to ensure reliable operation of internal components.
- Mobile units docked or connected to AC power versus battery power (DC).

Many of the projected uses for PC 98 computers include CPU-intensive and memory-intensive activities that further stress the power demands on mobile PCs.

Mobile PC Design Requirements

This section summarizes the additional design exceptions and design requirements for mobile PCs beyond those defined for Basic PC 98.

1. Mobile PC performance meets PC 98 minimum requirements
Required

For mobile PC systems based on Intel Architecture processors, minimum PC 98 performance requirements include the following:

- Minimum required CPU performance is 166-MHz Pentium processor with MMX technology, or equivalent performance.
- Minimum required performance is an L2 cache with 256K, or equivalent performance.
- Minimum required system memory is 24 MB. The Basic PC 98 limitations for memory available to the operating system apply for mobile PCs.

  Recommended: 32 MB RAM.

For mini-notebook requirements, see the "Mini-notebook Guidelines" section later in this chapter.

2. Mobile PC supports Smart Battery or ACPI-specified battery
Required

Recommended: Smart Battery.

If Smart Battery is implemented, the battery must meet the requirements defined in the *Smart Battery System Specification*, and the charger must comply with the *Smart Battery Charger Specification, Version 1.0*, both of which are available at http://www.sbs-forum.org.

If a multiple-battery system using Smart Battery is implemented, the system must have a battery selector that complies with *the Smart Battery Selector Specification, Version 1.0*, also available at http://www.sbs-forum.org.

An embedded controller or System Management Bus (SMBus) interface is required on systems that contain a Smart Battery solution, as described Section 13 of the ACPI 1.0 specification.

An ACPI Control Method Battery, defined in Section 11 of the ACPI 1.0 specification, also meets the PC 98 battery requirement.

3. Expansion capabilities of mobile PC are accessible to users

Required

Expansion capabilities in a mobile PC usually require external connections and, occasionally, additional internal components. The expansion slot is almost always physically blocked by access doors. Such doors are recommended for traveling integrity and to minimize entry of dust. This requirement is met if the user can access such external expansion slots without tools.

Internal expansion capabilities that require internal replacements, such as CPU, memory, built-in modem, and so on, are exempt from this requirement.

4. Mobile PC connections use icons plus keyed or shrouded connectors

Required

This requirement is the same as for Basic PC 98, except that for mobile PC designs, with small-height considerations, connector icons might not fit on the back of the case. In such cases, it is acceptable to wrap the icons to the bottom of the unit or place them on the inside of an access door.

5. Mobile PC includes a USB port

Required

For mobile PCs, a USB port must be built into the PC, not provided solely by port replicators or docking stations, although these units can provide extra USB connectors.

6. Mobile PC includes an IEEE 1394 port

Recommended

A docking station should include at least one IEEE 1394 port, as defined in the PC 98 system requirements in the "Basic PC 98" chapter in Part 2 of this guide. If implemented on a mobile PC unit, the IEEE 1394 port must comply with the requirements defined in the *1394 Device Power Management* specification when the final draft is approved by the 1394 Trade Association. For more information about power management requirements for IEEE 1394, see the "IEEE 1394" chapter in Part 3 of this guide.

7. USB-connected device does not maintain fully on power state

Required

An internal device that connects to the mobile PC using USB must not continually maintain the system in a fully-on power state. Such a device will override system power-management settings that control power-saving modes to protect battery life. When any USB device is connected but not active, the driver must allow system power management to suspend the notebook.

8. Mobile PC includes CardBus

Required

At least one 32-bit Type-2 CardBus slot (not 16-bit) is required. Additionally, Zoomed Video (ZV) support is recommended.

Note: Each device in a multifunction add-on device—such as a CardBus card—must separately meet the power management device class specifications for its device class and be independently power managed. This means that both device A and device B on the same add-on card do not have to be idle before the devices can be power managed. For information, including all requirements for CardBus support, see the "PC Card" chapter in Part 3 of this guide.

9. Mobile PC keyboard and pointing device meet PC 98 requirements

Required

The internal keyboard and any built-in pointing devices such a mouse, stylus, pen, touch pad, touch screen, trackball, stick, and so on required for a mobile PC should use standard system-board devices. The USB port can be used to support the requirement for external pointing device and keyboard connections. Alternatively, two PS/2-style ports can be implemented for the pointing device and keyboard, or a single PS/2-style port can be provided for both the pointing device and the keyboard.

For more information, see the "I/O Ports and Devices" chapter in Part 4 of this guide, which also provides information about implementing the recommended Windows and application logo keys on mobile PCs.

10. Mobile PC includes wireless capabilities

Recommended

If implemented, IrDA fast IR is recommended for synchronizing data exchanges with new peripherals such as digital still cameras. In addition, the software must have access to turning off the interface (D3 power state) using bus-specific methods or the methods defined in Section 3.4 of the ACPI 1.0 specification.

Standards for wireless PC peripherals are being developed within IrDA, with completion expected in 1997. For information, see the "I/O Ports and Devices chapter in Part 4 of this guide.

11. Mobile PC includes support for installing the operating system

Required

For mobile PCs, it is recognized that the system as purchased might not include all peripherals required for operating-system installation. Therefore, the Basic PC 98 system support for user installation of the operating system is required, but the user might need to access another PC 98 computer using a serial, parallel, or network connection to complete installation.

12. Mobile PC audio meets PC 98 audio requirements

Recommended

If audio is implemented in a mobile PC system, it must meet the requirements for PC 98 audio as defined in the "Audio Components" chapter in Part 4 of this guide.

13. Mobile PC includes communications device

Recommended

The recommended communication devices and guidelines are the same for mobile PCs as for Basic PC 98. The exception for modems is that if modem capabilities are integrated in the base platform, then V.80 or better is required. If modem capabilities are not integrated in the base platform, then V.80 is recommended. The exception for network adapters is that support is optional for remote new system setup capabilities as defined in the "Network Communications" chapter in Part 4 of this guide.

Notice that the presence of a CardBus slot on the mobile PC meets the Basic PC 98 requirement for providing either a modem or network adapter with a Mobile PC 98 system.

14. Built-in display adapter meets PC 98 minimum requirements

Required

The minimum required resolution for the built-in display is 800 × 600 × 16 bpp or 1024 × 768 × 8 bpp.

Important: For mobile PCs, most Basic PC 98 graphics requirements apply. However, the following are recommended rather than required for mobile PCs:

- Compliance with VESA 60-Hz noninterlaced refresh rate.
- General support for multiple adapters and multiple monitors.

  Notice that if a docking station is implemented, the base unit BIOS must have support for multiple adapters and multiple monitors as defined in the "Graphics Adapters" chapter in Part 4 of this guide. This support allows a user to add a graphics adapter in the docking station.
- MPEG-2 and DVD-Video support features.
- 3-D features to accelerate texture mapping, lighting, and so on.
- Accelerated Graphics Port (AGP).

15. Mobile system supports hot pluggable devices and alternative server connections

Recommended

For a mobile system, the following are additional design considerations:

- Supporting hot-pluggable devices that do not require a system reboot for insertion or removal.
- Including alternative methods for server connection because a LAN or dialup connection might not always be available. Methods can include a floppy boot disk, PC Card network adapter, LAN on the system board, or docking to support remote new system setup.

Support for remote wake up is not required for mobile PCs running on battery power. A CardBus implementation that supports the power management event (PME) signal meets this requirement, whether or not cards are available for testing. For information about PME signal definition, see *PCI Bus Power Management Interface Specification for PCI to CardBus Bridge, Revision 1.0* or higher.

Docking Station Requirements

Mobile PC docking systems allow docking of a PC, with additional hardware capabilities. A docking station allows the end user to add other devices to the mobile PC system—for example, sound, network adapter, hard disks, CD-ROM, different display adapter, SCSI, modems, and so on.

Docking systems can support hot, warm, or cold docking. Warm docking refers to docking and undocking the mobile PC while the system is in a low power state (as defined in the ACPI 1.0 specification) but is not powered off. Hot docking refers to docking and undocking the mobile PC while the system is operating at full power and is in an active working state.

Resource conflicts can occur when a mobile PC is paired with a docking station that allows users to add non-proprietary expansion cards to the system. For a mobile PC and docking station pair, the system designer must ensure that the docking system is capable of arbitrating resources for conflicts that might occur if an expansion card is added to the docking station. However, the system designer does not need to add to the mobile PC unit all of the Basic PC 98 resource-arbitration capabilities.

The requirements in this section apply for mobile designs that include a docking station. There is no requirement that a mobile PC must have a docking station.

16. Mobile PC/docking station combination meets PC 98 requirements

Required

Manufacturers must submit the combined docking station and mobile PC for PC 98 compatibility testing, and this combination must pass testing.

The docking unit must be able to power the mobile system and charge the mobile system's battery under the control of the mobile system.

Some PC 98 requirements might apply to a mobile PC/docking station combination that do not apply to the mobile PC as a standalone unit. The intent for PC 98 is that such requirements apply only because of facilities present in the docking station. For example, if a docking station provides graphics capabilities that substitute for the graphics capabilities of the mobile unit, the Basic PC 98 graphics requirements apply for the mobile PC/docking station combination when the substituted graphics component is in use. If the mobile PC is supplying all graphics capabilities, then Mobile PC 98 graphics requirements still apply.

17. Docking station meets all Basic PC 98 requirements

Required

The PC 98 requirements, as defined in the "Basic PC 98" chapter in Part 2 of this guide, include requirements for OnNow and ACPI, Plug and Play, and bus and device specifications.

All PC 98 Plug and Play requirements must be met if the docking station allows addition of non-proprietary devices. Complete compliance is not necessary if the docking station does not allow addition of non-proprietary devices.

The docking station must meet the PC 98 BIOS requirement for multiple adapters and multiple monitors, which allows for the graphics capabilities in the mobile unit to be fully operational (either the LCD panel or external connector) in the event that an user adds another graphics adapter to the docking station.

Many docking stations support VCR-style docking in which the notebook is closed when docked, so the user is prevented from accessing the notebook display. It is recommended that users not be precluded from accessing their notebook display when docked and that users have the option of simultaneously using the main display on the docking station and the notebook display.

Note: The docking station can support expansion capabilities through user-accessible ISA connectors, although such designs are discouraged for PC 98. It is expected that 1998 is the last year ISA will be allowed for end-user expansion in docking stations.

18. Docking station interface is supported using ACPI-defined mechanisms

Required

The docking station interface must be implemented using mechanisms defined in the ACPI 1.0 specification. Non-Plug and Play devices are enumerated using ACPI. All notification events and docking control must be implemented as defined in Sections 5.6.3 and 6.3 of the ACPI 1.0 specification.

19. Mobile PC/docking station combination supports automatic resource assignment and dynamic disable capabilities

Required

The mobile PC unit that is part of a docking system does not require all of the resource-arbitration capabilities required for expandable PC systems. However, the system as a whole must be capable of completely and dynamically disabling add-on devices and of freeing all the resources used by that device when the mobile unit is docked. This requirement excludes fixed-resource devices such as the DMA controller, interrupt controller, and so on.

With this capability, individual devices in the mobile PC will be disabled when it is docked, allowing the appropriate devices in the docking station to be enabled.

The system could fail if an add-on card requires resources that conflict with a device on either the mobile PC or the docking station. The mobile PC/docking station combination must be able to resolve resource conflicts among all the devices in the docking system.

This means that docking station devices must be available to replace disabled devices in the mobile PC, and these devices must meet the basic Plug and Play resource arbitration requirements for PC 98, as described in the "Basic PC 98 General Device Requirements" section in the "Basic PC 98" chapter. However, it is up to the design engineer of a mobile PC/docking station combination to determine which component (mobile PC or docking station) will resolve the conflict when the mobile unit is docked.

For more information about resource arbitration when two devices such as two keyboards or two mice are present, see the "Automatic resource assignment and dynamic disable capabilities are supported" requirement in the "I/O Ports and Devices" chapter in Part 4 of this guide.

20. Docking station supports warm docking

Required

Recommended: Support hot docking.

Docking or undocking a mobile unit from a docking station must not require powering off the system and must not require a system reboot.

Removable IDE devices are not required to meet this requirement.

21. Docking system supports fail-safe docking

Required

The system must provide a fail-safe system for docking and undocking the mobile unit. Working in conjunction with the operating system and ACPI (as defined in Sections 6.3 and 5.63 of the ACPI 1.0 specification), the mechanism for fail-safe docking must ensure the following:

- The undock button signals the user's intent to the system.
- Docking can occur only when the mobile unit is in the correct power state. The power state depends on whether the system is designed to support cold, warm, or hot docking.
- The user can initiate undocking through Windows-based software choices. Notice, however, that a hardware "button" must also be provided, because experience shows that users often do not find the software option and remove mobile units without operating system notification.
- The undock button or software choice sends a signal to the operating system so that the user is warned if resources are in danger of being lost.
- A safe-undock indicator is provided so the user can identify when it is safe to remove the mobile unit. This can be an LED or any other mechanism chosen by the vendor. If a physical mechanism automatically undocks the mobile PC or if hot docking is supported, then the safe-undock indicator is not required.

There is no requirement for mechanical lockout to block the user from removing the mobile unit without operating-system notification.

Port Replicator Requirements

A port replicator duplicates externally and extends features that are already available in a mobile PC—for example, an extra PC Card slot or keyboard and monitor connectors.

A port replicator with dedicated features allows the end user to add a specific feature to the original mobile PC—for example, a CD-ROM drive.

A mobile PC with a port replicator does not need to meet the expansion card requirements and does not need to meet all the resource requirements of a mobile PC/docking station combination. A port replicator is not required to provide an undock or eject button.

However, some mobile PC system designs include a port replicator that has dedicated features for networking, additional PC Card slots, a CD-ROM, and so on. This means that the system could have additional resource requirements to the point that all available IRQs in the system are already allocated; in this case, the PC Card slots (for example) would not have any IRQs available, rendering them useless.

In such cases, the port replicator must contain devices that replace any devices in the mobile PC that do not meet the IRQ, DMA, I/O port, and memory requirements for PC 98. This allows the operating system to disable the device on the mobile PC, to enable the corresponding device on the port replicator, and then to arbitrate resources among the remaining devices in the mobile unit and on the port replicator.

The requirements in this section apply for any port replicator designed for a PC 98 mobile PC. There is no requirement that a mobile PC must have a port replicator.

22. Port replicator supports automatic resource assignment and dynamic disable capabilities for replacement devices

Required

A port replicator that can accept expansion cards must contain devices that replace any devices in the mobile PC that do not meet Basic PC 98 requirements for IRQ, DMA, I/O port, and memory resources. This allows the operating system to disable the device on the mobile PC, to enable the corresponding device on the port replicator, and then to arbitrate resources among the remaining devices in the mobile unit and on the port replicator.

Devices in the system must be capable of being dynamically disabled so that the user can choose to free resources in order to allow other devices in the system to function.

Tip: To avoid resource shortages, the system designer can take advantage of the capability of Yenta-compliant CardBus controllers' capability to assign a shared PCI interrupt for R2 PC Cards, rather than using IRQs, as defined in the "PC Card 16 card driver supports sharing of level-mode interrupts" item in the "PC Card" chapter in Part 3 of this guide. For more information, see the related article at http://www.microsoft.com/hwdev/busbios/.

23. Port replicator supports warm docking

Required

Docking or undocking a mobile unit from a port replicator must not require powering off the system and must not require a system reboot.

Removable IDE devices are not required to meet this requirement.

Mini-notebook Guidelines

This section summarizes specific requirements for mini-notebook mobile PCs. All requirements in this chapter must be met by mini-notebooks unless an exception is specifically defined in this section.

For PC 98, a mini-notebook is defined as a system that has a carry weight of 3 pounds or less, including all hardware required to run the Windows operating system.

24. Mini-notebook performance meets PC 98 minimum requirements
Required
For mini-notebook systems, the minimum PC 98 performance requirements consist of the following:

- Minimum required CPU performance is 133-MHz Pentium processor with MMX technology, or equivalent performance.
- Minimum required system memory is 16 MB. The Basic PC 98 requirements apply: no more than 4 MB of system memory can be locked and unavailable to the operating system.
- Minimum required display is 640 × 480 × 8 bpp. Compliance with 15-bpp or 16-bpp specifications is recommended.
- System includes all functionality required to run the Windows operating system.

All other Mobile PC 98 and Basic PC 98 requirements beyond those listed here as the minimum requirements are optional for mini-notebooks.

Mobile PC 98 References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

1394 Device Power Management
ftp://ftp.p1394pm.org/pub/1394pm/
http://www.microsoft.com/hwdev/onnow.htm

Advanced Configuration and Power Interface Specification, Revision 1.0
http://www.teleport.com/~acpi/

El Torito—Bootable CD-ROM Format Specification, Version 1.0
Compaq, Intel, Phoenix BIOS Boot Specification, Version 1.01
http://www.ptltd.com/techs/specs.html

Intel hardware developer site
http://developer.intel.com

PCI Bus Power Management Interface Specification for PCI to CardBus Bridge, Revision 1.0
http://www.pcisig.com

Plug and Play specifications
http://www.microsoft.com/hwdev/specs/

Power management specifications for device and bus classes
Guidelines for audible noise and other OnNow technologies
http://www.microsoft.com/hwdev/onnow.htm

Smart Battery Charger Specification, Version 1.0
Smart Battery Selector Specification, Version 1.0
http://www.sbs-forum.org

Windows and Windows NT DDKs
MSDN Professional membership

Checklist for Mobile PC 98

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

1. Mobile PC performance meets PC 98 minimum requirements
Required

2. Mobile PC supports Smart Battery or ACPI-specified battery
Required

3. Expansion capabilities of mobile PC are accessible to users
Required

4. Mobile PC connections use icons plus keyed or shrouded connectors
Required

5. Mobile PC includes a USB port
Required

6. Mobile PC includes an IEEE 1394 port
Recommended

7. USB-connected device does not maintain fully on power state
Required

8. Mobile PC includes CardBus
Required

9. Mobile PC keyboard and pointing device meet PC 98 requirements
Required

10. Mobile PC includes wireless capabilities
Recommended

11. Mobile PC includes support for installing the operating system
Required

12. Mobile PC audio meets PC 98 audio requirements
Recommended

13. Mobile PC includes communications device
Recommended

14. Built-in display adapter meets PC 98 minimum requirements
Required

15. Mobile system supports hot pluggable devices and alternative server connections
Recommended

16. Mobile PC/docking station combination meets PC 98 requirements
Required

17. Docking station meets all Basic PC 98 requirements
Required

18. Docking station interface is supported using ACPI-defined mechanisms
Required

19. Mobile PC/docking station combination supports automatic resource assignment and dynamic disable capabilities
Required

20. Docking station supports warm docking
Required

21. Docking system supports fail-safe docking
Required

22. Port replicator supports automatic resource assignment and dynamic disable capabilities for replacement devices
Required

23. Port replicator supports warm docking
Required

24. Mini-notebook performance meets PC 98 minimum requirements
Required

PART 3

PC 98 Bus Design Guidelines

C H A P T E R 7

USB

This chapter presents the PC 98 requirements and recommendations for Universal Serial Bus (USB).

USB provides an expandable, hot-attachable Plug and Play serial interface for adding external peripheral devices ranging from interactive HIDs such as joysticks and pointing devices to isochronous devices such as telephony, audio, and imaging devices. USB allows cascading hubs that can be integrated into desktop devices such as monitors and keyboards.

For PC 98, USB provides a standard, low-cost socket that accommodates volume emerging and legacy I/O devices. This feature is required on all PCs, and migration of I/O devices from legacy ports to USB is recommended. In particular, the joystick, pointing device, and keyboard devices that ship with PC systems should be USB.

For Windows and Windows NT support, devices can use the generic class drivers provided with the operating system, or manufacturers can create drivers or WDM minidrivers (depending on the device class) to exploit any additional unique hardware features. For details, see the “I/O Ports and Devices” chapter in Part 4 of this guide.

Manufacturers should ensure that their USB devices are tested at the compatibility workshops provided by the USB Implementers Forum.

Contents

USB Basic Requirements

This section summarizes the basic USB design requirements.

1. USB included on PC 98 system

Required

This is required for all PC 98 system types. As noted in the “Basic PC 98” chapter in Part 2 of this guide, BIOS boot support is required when a USB keyboard is the sole keyboard support provided with the PC system.

2. All USB hardware complies with USB 1.0 specification

Required

All USB hardware must comply with *USB Specification, Version 1.0* or higher, published by Compaq, Digital Equipment, IBM, Intel, Microsoft, NEC, and Nortel. This ensures that USB hardware has complete Plug and Play capabilities and is implemented in a standard way.

For example, on any system with USB capabilities, a user must be able to dynamically attach any USB peripheral to any USB connector. The operating system should automatically recognize it, load and initialize the appropriate drivers, and make the device available for use.

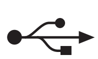
3. Connections use USB icon

Required

The icon can be molded, printed, or affixed as a permanent sticker. Because the location and number of USB ports can be vary, appropriate icons on both ports and cables are important ease-of-use factors. Therefore, USB icons are required for external cables, connecting cables, and connection ports.

Icons can be based on vendor designs, or vendors can use the recommended USB icon defined in Chapter 6 of the USB 1.0 specification and illustrated here:

> The USB icon should be molded into the connector and also placed on the product for ease of identifying the USB port. It is recommended that the icon on the product and the one on the plug be adjacent to each other when the plug and receptacle are mated. This icon can be used for both series A and B connector schemes. On the plug, there should be a 0.635-mm rectangular recessed area around the icon such that there is a perceptible feel of the icon.



4. Devices and drivers support maximum flexibility of hardware interface options

Recommended

Device and driver designs must provide maximum flexibility of interface options in order to allow user-preference coordination by the operating system or other resource managers. This will allow graceful use of multiple simultaneous devices and applications in a dynamic environment.

Specifically, devices with configurations or interfaces that contain isochronous endpoints should not consume any USB bandwidth when the device is first configured. This can be done by having the zero AltSetting for any interface consume no bandwidth. When the device is put into operation, the device driver should switch the device to an AltSetting that allocates and consumes the required amount of bandwidth. When the device is no longer being used, the driver should return the device to an AltSetting where bandwidth is not consumed.

USB Host Controller Requirements

This section summarizes USB class specifications and standards for host controllers.

5. USB host controller meets either OpenHCI or UHCI specification

Required

The host controller must be compliant with the specifications for either OpenHCI (Open Host Controller Interface; published by Compaq, Microsoft, and National Semiconductor) or UHCI (Universal HCI; published by Intel). Hardware manufacturers who design to one of these specifications are not required to provide an additional device driver for their host controller under the Windows or Windows NT operating systems.

Multiple OpenHCI and UHCI USB controllers are supported concurrently by the operating system.

6. USB host controller can wake the system

Required

The USB host controller must support wake-up capabilities from at least one of the S1, S2 or S3 system sleep states.

USB Power Management

This section summarizes the specific USB power management requirements.

7. System and devices comply with USB power management requirements
Required

PC 98 systems and devices must implement the power descriptor in the USB 1.0 specification. Complete implementation guidelines for OnNow and USB are defined in the "OnNow requirements in the USB Core Specification" section of the article titled "OnNow Power Management and USB" on the web site at http://www.microsoft.com/hwdev/pcfuture/.

Design Features for USB Peripherals

This section summarizes requirements related to bus-class specifications and standards for peripherals that use USB.

8. USB devices meet requirements in related USB device class specification
Required

Every device must comply with the *USB Common Class Specification, Version 1.0* or higher. For any add-on device or peripheral that fits into one of the USB device class definitions, the device must comply with the related USB device class specification. USB class drivers in the operating system are implemented to support compliant devices in each particular class. Class driver extensions and WDM allow IHVs to innovate and differentiate their products while still meeting class compliance in their base operational modes.

USB References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

Intel information about USB, including the UHCI design guide for USB
http://developer.intel.com/design/litcentr/litweb/usb.htm
http://developer.intel.com/design/usb/

OnNow Power Management and USB and other OnNow-related articles
http://www.microsoft.com/hwdev/onnow.htm

Open Host Controller Interface, Version 1.0
http://www.microsoft.com/hwdev/specs/

USB Class Definition for Communications Devices, Version 0.9
USB Common Class Specification, Version 0.9
USB Device Class Definition for Audio Devices, Version 0.9
USB Device Class Definition for Human Interface Devices, Version 1.0
USB Device Class Definition for Mass Storage Devices, Version 0.9
USB Device Class Definition for Printing Devices, Version 1.0
USB HID Usages Tables, Version 0.9
USB Monitor Control Class Specification, Version 1.0
USB Power Devices Usages Table, Version 0.9
USB Specification, Version 1.0
USB Implementers Forum
Phone: (503) 264-0590
Fax: (503) 693-7975
http://www.usb.org

Windows NT DDK
MSDN Professional membership

Checklist for USB

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

1. USB included on PC 98 system
Required

2. All USB hardware complies with USB 1.0 specifications
Required

3. Connections use USB icon
Required

4. Devices and drivers support maximum flexibility of hardware interface options
Recommended

5. USB host controller meets either OpenHCI or UHCI specification
Required

6. USB host controller can wake the system
Required

7. System and devices comply with USB power management requirements
Required

8. USB devices meet requirements in related USB device class specification
Required

CHAPTER 8

IEEE 1394

This chapter summarizes PC 98 design requirements for hardware designed under the IEEE 1394 standards. The IEEE 1394 high-speed serial bus complements USB by providing enhanced PC connectivity for a wide range of devices, including consumer audio/video (A/V) components, storage peripherals, other PCs, and portable devices.

IEEE 1394 has been adopted by the consumer-electronics industry and is expected to provide a volume, Plug and Play-compatible expansion interface for the PC. The 100-Mb/s, 200-Mb/s, and 400-Mb/s transfer rates currently specified in IEEE 1394A, and the proposed enhancements to 800 Mb/s and beyond in IEEE 1394B, are well suited to multistreaming I/O requirements.

Contents

IEEE 1394 Basic Requirements

The following is a summary of the IEEE 1394 design considerations related to PC systems, as addressed in this chapter:

- Compliance with various approved IEEE 1394 standards
- Support for the OpenHCI specification for controllers
- Plug and Play support for device configuration, control and status registers (CSRs), connectors and cabling, and connection fault handling
- Cable power distribution, including requirements for source devices, sink devices, self-powered devices, and supporting CSRs
- Device power management, CSRs, and soft-power protocols
- Device command protocols for audio, video imaging, still imaging, and storage device classes

This section defines the basic PC 98 requirements for IEEE 1394.

1. Controllers and devices support IEEE 1394-1995 standards
Required
Designs that interface to the IEEE 1394 bus must support the following industry standards and supplemental specifications:

- IEEE 1394-1995 standard
- IEEE 1394A, an amendment to IEEE 1394-1995
- IEEE 1212-1991 CSR Format (ISO/IEC 13213:1994)

2. Controllers comply with OpenHCI for IEEE 1394
Required
The OpenHCI specification for IEEE 1394 defines standard hardware and software for PC connections to the IEEE 1394 bus. OpenHCI defines standard register addresses and functions, data structures, and DMA models. The benefits of this standard include improved performance, security, and error handling.

An IEEE 1394 OpenHCI device is bus manager-capable, including bus mastering for BANDWIDTH_AVAILABLE and CHANNELS_AVAILABLE registers.

3. OpenHCI controllers and devices support advances defined in IEEE 1394A
Required

The advances in the IEEE 1394A specification enhance system performance and integration of component systems.

4. Host supports peak data rate of 400 Mb/s, minimum
Required

The integration of component systems that enable concurrent applications demands minimum bandwidth for an effective user experience. A peak data rate of 400 Mb/s is required for effective integration of systems and devices targeted for use with PCs in 1998.

5. Design avoids excessive currents resulting from ground-fault potential among devices
Recommended

PC-based peripherals are not required to implement isolation because of the usual assumption of a common green-wire ground for all linked devices. Accordingly, the requirement for electrical isolation has been targeted for removal from the IEEE standard.

For local area network (LAN) configurations, it is desirable to avoid excessive currents resulting from ground-fault potential among devices. The related design problem can be solved for such configurations by building isolation into the power supply and by AC coupling of the physical layer device (PHY) and Link interface of selective AC-powered subsystems.

Notice that a mobile device powered by AC is considered to be an AC device, even with a battery (DC) present. The PC uses this level of definition to budget cable power.

Requirements for IEEE 1394 Devices

This section summarizes additional requirements for IEEE 1394 peripherals such as consumer-electronics devices.

6. Device command protocols conform to standard device class interfaces
Required

For complete information, see the WDM reference in the Windows NT 5.0 DDK.

7. Devices support peak data rate of 400 Mb/s, minimum
Recommended

For PC 98 designs, 400-Mb/s IEEE 1394 devices are strongly recommended; 100-Mb/s devices are strongly discouraged; and 200-Mb/s devices should limit their peak bus utilization to less than 50 percent.

For nonhost devices, a 200-Mb/s device with a 160-Mb/s data stream requires 80 percent bus utilization, effectively lowering overall bus bandwidth to 200 Mb/s for 80 percent of the time. Therefore, low bus utilization is required in order for 100-Mb/s and 200-Mb/s devices to coexist with 400-Mb/s peripheral devices. For example, three devices performing at 200 Mb/s each with 30 percent bus utilization would saturate the bus.

Also, application bandwidth can be limited by speed traps (that is, a slow device separating two faster devices), imposing speed-dependent cabling considerations on the end user.

8. Devices requiring support for high-bandwidth data transfer use IEEE 1394
Recommended

For devices that require support for high-bandwidth data transfers and Plug and Play connectivity, the IEEE 1394 bus is recommended. Such devices include the following:

Component audio	DVD	Printers
Digital camcorder	Hard disk drives	Set-top television controllers
DTV	High-resolution scanners	Video conferencing cameras
Digital VCR	PC docking stations	

Plug and Play for IEEE 1394

This section summarizes the Plug and Play requirements for IEEE 1394 peripheral devices and PC host controllers.

9. Plug and Play devices demonstrate interoperability with other devices
Required

All devices must support Plug and Play for intended applications in both a minimal and an extended bus configuration. A minimal configuration is the minimum number of devices necessary to demonstrate the primary application of the device. An extended configuration is an advanced application with at least two devices added to the minimal configuration. The added devices can be extraneous to the application.

The following is a summary of compliance testing guidelines for this requirement:

- Intended applications must be documented before testing.
- Both test configurations must consist of a core matrix of stable devices that have demonstrated full interoperability in the absence of the test device. To be included in the core test matrix, a device must have demonstrated compliance of its PHY, Link, and Transaction layers as specified in the IEEE 1394-1995 standard.
- The core matrix of devices must be established by an independent agency such as the 1394 Trade Association, with actual testing performed by an independent third party or as part of an industry compatibility workshop.

10. Topology faults do not cause the bus to fail
Required

Standard IEEE 1394 protocols have been defined to eliminate topology faults. However, to ensure correct implementation, the following items describe test criteria for industry compatibility workshops. In each case, connection or removal of a device must not stall the bus, but the faulting device might not function. The PC must detect each fault. The test criteria include the following:

- Surprise removal. All isochronous-capable devices must support the Connection Management Protocol specified in IEC 1883 (or the most recent specification) in order to resume streaming connections following a bus reset and to de-allocate channels upon surprise removal of a device.
- Safe removal. All devices that provide a front-panel power switch must signal the operating system in response to a local shut-down request (that is, hot unplugging) in order to allow safe removal. Safe removal requires that the end user monitor the PC bus manager’s response to the request before removing the device.

- Greater than 16 cable hops. If the user extends the bus beyond 16 hops (that is, device-to-device daisy-chain connections), the total distance end-to-end approaches 16 × 4.5 meters = 72 meters. This distance exceeds propagation delay times for fair arbitration timing, potentially starving a node on a heavily loaded bus. Cable lengths must not exceed 4.5 meters. In this scenario, the PC is acting as bus manager and must detect the topology fault and provide a warning message to the user.
- Greater than 63 devices on a local IEEE 1394 bus. If the 63-device limit is exceeded, the 64th and later devices will be assigned a physical ID of 63. The 64th device must be detected by the PC bus manager and must provide a warning message to the user.

11. Removable media devices support media status notification

Required

Removable media devices must use an electronic switch to notify the PC in the event of media change requests. This is necessary to enable device applications to lock, unlock, and eject media.

12. Devices that can initiate peer-to-peer communications also support remote programming

Required

To enhance systems integration, all devices capable of initiating peer-to-peer communications must also support a programming language that enables remote control for PC applications. This allows a third device, such as a PC or device controller, to initiate data transmission between two devices.

Plug and Play for Device Configuration ROM

This section defines the Plug and Play requirements related to device configuration ROM.

13. Device provides a configuration ROM for unique device identification

Required

For Plug and Play device control, the device configuration ROM must provide configuration information as specified in the IEEE 1394-1995 standard and as outlined in Table 1 (see following). The configuration ROM is required for unique detection of the device and is used by a PC to enumerate the bus and to load the correct device driver. Table 1 provides an example ROM that combines all the elements outlined in the requirements listed in this section. For up-to-date information about the configuration ROM, see the web site at http://www.microsoft.com/hwdev/busbios/.

Table 1. Configuration ROM (located at FFFF F000 0400)

Block	Offset	Description			
First Quadlet	400h	bus_info_length 04h	CRC_length 17h	ROM_CRC_value (calculated)	
Bus_Info Block	404h	'1394' in ASCII 31h	33h	39h	34h
	408h	m c i b p reserved	cyc_clk_acc	max_rec	reserved
	40Ch	node_vendor_id			chip_id-hi
	410h	chip_id-low			
Root Directory	414h	Directory Length 00h	04h	Directory CRC (calculated)	
	418h	vendor_ID key 03h	module_vendor_id		
	41Ch	nod_ capabilities key 0Ch	node_capabilities 00h	83h	80h
	420h	node_unique_id key 8Dh	node_unique_id leaf offset 00h	00h	02h
	424h	unit directory key D1h	unit directory offset 00h	00h	04h
Node Unique ID Leaf	428h	Length of leaf 00h	02h	Leaf CRC (calculated)	
	42Ch	node_vendor_id			chip_id_hi
	430h	chip_id_lo			
Unit Directory	434h	Unit Directory Length 00h	06h	Directory CRC (calculated)	
	438h	Unit spec key 12h	unit_spec_id		
	43Ch	unit_sw_version key 13h	unit_sw_version 01h	04h	83h
Unit Directory (battery)	450h	Unit Directory Length 00h	03h	Directory CRC (calculated)	
	454h	Unit spec key 12h	53h	44h	50h
	45Ch	Unit sw version key 13h	00h	00h	01h
	460h	Battery status reporting key 77h	BATTERY_STATUS_REPORTING Offset		

14. Device configuration ROM implements general ROM format

Required

The general configuration ROM format is specified in the IEEE 1394-1995 and ISO/IEC 13213:1994 standards. The general ROM format is an extensible tree structure that enables a managed environment by providing node-specific and unit-specific information as required for Plug and Play, power management, and isochronous data transfers. The general ROM format also provides for definition of multifunction device units. The bus information block and root directory of the general ROM format are required as specified in Table 1.

15. Bus information block implemented at a base address offset of 0404h

Required

The format of the bus information block is defined by the IEEE 1394-1995 standard. The first quadlet of the bus information block at offset 404h is the configuration ROM signature field used to identify an IEEE 1394 configuration ROM. This quadlet must contain the ASCII string "**1394**". The second quadlet of the bus information block at offset 408h contains several bits that indicate node capabilities. These bits are defined as shown in the following table, together with their required values.

Bits Indicating Node Capabilities at Offset 408h

Bit or field	Table 1 symbol	Value and description
irmc bit	m	Must be 1. Indicates that the node supports isochronous resource manager capabilities.
cmc bit	c	Must be 1 if the node supports cycle master capabilities; otherwise, this value must be 0.
isc bit	i	Must be 1 if the node supports isochronous operations; otherwise, this value must be 0.
bmc bit	b	Must be 1. Indicates that the node supports bus manager capabilities.
pmc bit	p	Must be 1. Indicates that the node is power manager capable. The *pmc* bit is not defined by the IEEE 1394-1995 standard and is an extension created by this specification.
cyc_clk_acc field	—	Specifies the accuracy of the node's cycle master clock in parts per million. If the *cmc* bit is 1, the field's value must be between 0 and 100. If the *cmc* bit is 0, this field must be all ones.
max_rec field	—	Defines the maximum payload size of a block-write transaction addressed to the node. The range of the maximum payload size is from 4 to 2048 bytes. A *max_rec* value of 0 indicates that the maximum payload size is not specified. Otherwise, within the range of defined payload sizes, the maximum size is equal to $2^{max\_rec + 1}$. The *max_rec* field does not place any limits on the maximum payload size in asynchronous data packets—either requests or responses—that the node might transmit.

16. Configuration ROM provides globally unique device ID

Required

The third and fourth quadlets of the bus information block of the configuration ROM must provide a globally unique device ID, which appears in Table 1 beginning at offset 40Ch. This unique 64-bit node ID is the only way to recognize the presence of a given device, because the physical device addresses can change following a bus reset. The unique ID is required for device detection and PC device driver loading.

If a bus node supports multiple units, then the unique 64-bit ID must not be referential to any one unit directory in order to allow for unique identification of a unit in a multifunction device.

The globally unique device ID in the bus information block must be invariant when read with quadlet read requests. That is, it must not be alterable in any way by software.

17. Root directory is located at a fixed address following the bus information block

Required

The root directory must be located at a fixed address following the bus information block. For example, the root directory shown in Table 1 is fixed at offset 414h. All other directories and leaves are addressed by entries in their parent directories starting with the root directory. The root directory contains pointers to the root-dependent directory, a node-power directory as specified in *1394 Specification for Power Management*, and unit directories for each independent device function.

18. Configuration ROM includes a unit directory for each independent device function

Required

A unit directory is required for independent function and control of each device unit. A valid pointer to a unit directory must be provided at offset 0x24h, as shown in Table 1, in compliance with the general ROM format specified in IEEE 1394-1995 and the directory format specified in ISO/IEC 13213:1994.

19. Each unit directory provides a valid Unit_Spec_Id and Unit_Sw_Version

Required

Within a unit directory, Unit_Spec_Id identifies the specification authority, and Unit_Sw_Version identifies the particular document describing the unit. When added to the beginning of Unit_Spec_Id, then Unit_Sw_Version uniquely identifies the unit’s software interface.

20. Each unit directory provides a pointer to a unit-dependent directory

Required

The unit-dependent leaf directory must provide additional information about the device unit's vendor and model in associated leaf directories. The format of the information contained in the vendor and model leaves is specific to Unit_Spec_Id and Unit_Sw_Version.

A valid pointer to a unit-dependent directory must be in accordance with the generic directory format specified in ISO/IEC 13213:1994. The unit-dependent directory must provide valid pointers to vendor and model leaves.

21. Vendor and model leaves support textual descriptor leaf format

Required

Textual descriptors are required for Unit_Spec_ID and Unit_Sw_Version entries in the configuration ROM in order to display this information to the user. Textual descriptors are recommended for all other configuration ROM entries. Each textual descriptor points to a leaf that contains a single character string.

Alternately, the textual descriptor can point to a directory that points to one or more textual descriptor leaves corresponding to supported languages. Leaf format and textual descriptor leaves are specified in ISO/IEC 13213:1994.

Textual descriptor leaves must include the following:

- The spec_type field must be "0" to correspond to a 24-bit specifier_id for a standards body, or "1" to correspond to a 24-bit specifier_id for a defining vendor company_id.
- The language_id field must be derived from the Windows NT locale number (a quadlet), OR'd with 0x80000000.
- Text string_info must be in ASCII for any language_id in the range 0–7fffffff or in Unicode for any language_id in the range 0x80000000–0xfffffff.

22. Unit-dependent directory provides a pointer to the unit's CSRs

Required

Each unit's CSRs must be in separate, non-overlapping address spaces to maintain independent device control. If CSRs can be used to interact with a device unit, the unit-dependent directory must provide a pointer to the base address of the unit's CSRs. This provides an easy way for an application or a device driver to access the unit's CSRs.

Plug and Play for Cabling and Connectors

This section defines the Plug and Play requirements for IEEE 1394 cabling and connectors.

23. Device provides three connector ports
Recommended

All devices should provide three 6-pin connector ports for optimum cabling options, subject to cable-power distribution constraints. Fewer than three ports promotes long daisy chains, increasing the potential for speed traps (a slow device separating two faster devices). Therefore, three-port IEEE 1394 device nodes are recommended, with exceptions noted in the "Device uses standard 6-pin IEEE-1394 connector" requirement later in this section.

For internal-only devices, a minimum of two ports enables daisy chaining of devices. However, a limit of 15 hops (end-to-end distance) restricts total devices to 16, sufficient for most internal configurations.

Devices that consume cable power should be limited to a single connector to encourage short source-to-sink power delivery while eliminating the build up of voltage drop associated with a long daisy chain of power consumers.

24. Device uses standard 6-pin IEEE 1394 connector
Required

A single connector eliminates unnecessary choices for the end user. For every *n* supported connector, there are 2^{n-1} cable choices. Two connector styles yield three end-user cable choices. Consistent use of the standard 6-pin IEEE 1394 connector eliminates an undesirable break in the power bus for power-dependent device applications.

Other benefits include volume pricing and consistent electrical performance. Therefore, all external pluggable IEEE 1394 devices must use the standard 6-pin IEEE 1394 connector. The exception is an option to use the 4-pin IEEE 1394A connector for miniature single-port (leaf-node) devices, as defined in the "Only single-port leaf-node devices use 4-pin connectors" requirement later in this section.

25. Self-powered devices propagate the power bus through each connector
Required

Self-powered devices provide their own power source and do not consume cable power. The exception to this rule is that a self-powered device can consume cable power up to the number of watts defined in *1394 Specification for Power Management* in order to power its own PHY if it is not able to continue to power its own PHY when the self-powered device has been turned off.

If the self-powered device does consume cable power to power its own PHY, it must always use cable power to do so whenever cable power is present. That is, it cannot dynamically switch between consuming and not consuming cable power for PHY power.

Self-powered devices that provide their own power source and do not consume cable power must maintain the electrical integrity of the power bus for other devices dependent on it. Therefore, all self-powered IEEE 1394 devices must propagate the power bus through each connector. To accomplish this, each self-powered device must short together the power pins and the ground pins of each connector.

26. Only single-port leaf-node devices use 4-pin connectors
Required

The 4-pin connector offers a slightly lower cost and a smaller footprint ideally suited to hand-held devices. Use of a unique leaf-node connector adds one more cable choice for end users. Therefore, devices can comply with this specification by restricting application of the 4-pin (powerless) A/V connector to single-connector leaf-node devices. The 4-pin connector is specified in the IEEE 1394A specification.

27. Device connectors exhibit common speed and power characteristics
Required

Devices with multiple connectors must exhibit common characteristics at each connector to reduce end-user cabling choices. All connectors on a device must exhibit homogeneous speed, power, and mechanical characteristics such that:

- Multiconnector devices use the 6-pin connector.
- All device connectors propagate the power bus.
- All device connectors support a common peak data rate.

Optionally, all devices providing cable power through 6-pin connectors must provide diode isolation as specified in the *1394 Trade Association Power Specification Part 1: Cable Power Distribution*.

28. Standard 400-Mb/s rated IEEE 1394 cable provided with devices

Required

For Plug and Play, it is important to use one standard-performance cable for all device configurations to eliminate cable choices for the end user. This is especially important given the range of devices possible on an IEEE 1394 bus. A mix of cable types and ratings creates an unfriendly user experience. Therefore, all cables must be have a minimum 400-Mb/s rating and, if bundled, must be shipped with a standard cable.

Plug and Play Power Interfaces

This section summarizes Plug and Play requirements for cable power distribution.

For Plug and Play, all devices—whether cable or self-powered—must comply with the applicable requirements in *1394 Specification for Power Management.* These requirements enable a power management-capable bus manager to provide instant-on application support while reducing system-wide device power consumption.

In addition, all devices must comply with the *1394 Trade Association Power Specification Part 1: Cable Power Distribution.* Although the requirements for devices that do not consume or produce cable power are minimal, all devices share responsibility for propagating the power bus as defined in the Cable Power Distribution specification.

A standard cable-power distribution model is necessary to reduce the likelihood of power-fault conditions, such as insufficient power for connection of a cable-powered device and surprise removal of a device power source. In addition, a bus manager that is power management-capable can allocate or de-allocate available power within diode-isolated power domains, accounting for the overall power budget and voltage drop.

Plug and Play requirements in this section highlight details specified in the applicable power specifications.

29. Devices power their PHY at all times

Required

All devices must perform the bus repeater function when powered down as specified in the IEEE 1394-1995 specification. Therefore, a device power switch must allow for local power to the PHY when switched off. Alternatively, a device can implement the standard protocol to request cable power (if available) from the power manager to power the PHY.

An exception to these requirements is necessary for PC add-on cards and system-board host connection devices that are subject to the power characteristics of the PCI bus.

A device that does not provide power to its PHY or consume power from the cable for its PHY will terminate the bus at the point of connection and must, therefore, terminate the pass-through of power.

30. Devices report power source and cable power consumption in Self_id packet

Required

Self-powered devices must report zero power consumed in the power class field of the Self_id packet. Alternately, if a device consumes cable power only to keep its PHY alive, it must report this consumption in the Self_id packet. This allows the power manager to reserve power for this occasion.

31. Devices implement link power control

Required

All cable-powered and self-powered devices must implement the Link_on packet and Link_off bit in the State_Clear register. These controls allow a power management-capable bus manager to control the node's power state. Access to the device configuration ROM must be possible following a Link_on. A device cannot increment its power consumption by more than 3 watts following a Link_on. Self-powered devices can power up with Link_on. However, cable-powered devices must rely on the power manager to enable their link.

32. Device requiring power increments in excess of Link_on implements unit-power CSRs

Required

All cable-powered and self-powered devices that require power increments in excess of Link_on power must implement standard unit-power CSRs as specified in *1394 Specification for Power Management.* This is necessary to allow for seamless integration of centralized power management capabilities when a device is connected to a mini-system.

In addition, all devices of a given device class must implement a standard set of unit power states as specified in the device class power management specification for that device class. For example, all VCRs must exhibit a consistent behavior with respect to power states and transitions between states. This is necessary to provide a consistent user experience.

Note: Please check with the 1394 Trade Association or send e-mail to 1394@microsoft.com to determine whether a power class specification exists for your device type. Alternatively, you are encouraged to draft a proposal for your device type and submit it to the 1394 Trade Association architecture working group for review and approval.

33. Devices that source cable power must report this capability

Required

This reporting is necessary to enable centralized power management. A device that sources 20 volts or more of cable power at 15 watts minimum must report that it provides power in its Self_id packet as specified in IEEE 1394-1995. Devices that provide less than 20 volts at 15 watts can be discovered using configuration ROM information as described in *1394 Specification for Power Management.*

34. IEEE 1394-enabled PC sources cable power

Required

An AC-powered PC must source cable power to the bus. Cable power in turn enhances Plug and Play with a single connection for low-cost cable-powered devices. Battery-powered mobile and notebook devices are exempt from this requirement, whether or not the device is connected to an AC adapter.

Minimum power wattage is defined in the following item.

35. Power source supplies a minimum of 20 volts at 15 watts

Recommended

To minimize the cost of a power source, actual power output can be reduced to less than 40 volts at 1.5 amps as specified in IEEE 1394-1995. Also, a cable power source should supply enough power for at least one cable-powered device (15 watts) while also addressing voltage drop in the cable. Therefore, a minimum cable power source of 20 volts DC at a current limit of 1 amp is recommended. However, at the expense of higher component ratings, a 30-volt cable power source will reduce power loss in the cable.

In addition, if the power provider specifies a power capability greater than 15 watts, it must be capable of providing that power under full load. A power provider is required to always be able to provide its stated power under full load conditions.

For example, a minimum 20 watts output will ensure delivery of only 15 watts to a load some distance away from the source device. This is because of a cable voltage drop of 5 volts—that is, 1 amp × 0.66 ohm × 7 to 8 cable hops separating source node from sink node at a rated cable-hop resistance of 0.66 ohms. The voltage at the load will drop to 15 volts, with the source current limited at 1 amp. Therefore, a practical design target for a cable power source is a minimum of 20 volts with a current limit of 1 amp.

A device such as a notebook that wants to source less than 20 volts can do so if it reports in its Self_id packet that it does not source power, but does report in its configuration ROM the exact power it provides.

36. Devices notify the power manager of power change requests

Required

All devices that produce or consume cable power must use an electronic power switch to notify the power manager of requests from the front panel to change the power state. This function must be accomplished using the notification request protocol specified in *1394 Specification for Power Management.* This protocol provides a time-out for defaulting to local control as is necessary for operation in non-power-managed environments.

This same mechanism is required for safe removal of a device (hot unplugging).

Power Management for IEEE 1394 Devices

All devices on the IEEE 1394 bus must comply with the power management requirements outlined in this section.

37. Devices and controllers comply with Cable Power Distribution specification

Required

The cable power distribution model has been defined to provide guidelines for implementation of devices that propagate, source, or sink cable power. Thus, all devices must satisfy power distribution requirements. *1394 Trade Association Power Specification Part 1: Cable Power Distribution* addresses interoperability and power distribution necessary for operation of both power-managed bus configurations and, with some restrictions, unmanaged bus configurations.

38. Devices and controllers comply with IEEE 1394 power specification

Required

Power-management CSRs and protocols provide an enhanced Plug and Play experience for end users. All devices must support power-state, power-capabilities, and power-status commands as defined in *1394 Specification for Power Management.* Cable-power devices must support the notification request protocol. Wake-up and battery-status CSRs are optional but strongly recommended.

IEEE 1394 References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

1394 Specification for Power Management
ftp://ftp.p1394pm.org/pub/p1394pm/
http://www.microsoft.com/hwdev/onnow.htm

1394 Trade Association
E-mail: 1394-sig@1394ta.org
http://www.1394ta.org

1394 Trade Association Power Specification Part 1: Cable Power Distribution
ftp:// ftp.p1394pm.org/pub/p1394pm/

IEC 1883 Digital Interface for Consumer Electronic Audio/Video Equipment
http://www.iec.ch

IEEE 1394 Standards
ASK*IEEE
Telephone: (800) 949-4333
Fax: (212) 310-4091
E-mail: askieee.ieee.org

Global Engineering Documents
Phone: (800) 854-7179 (US)
(613) 237-4250 (Canada)
(303) 792-2181 (Outside North America)
Fax: (303) 397-2740
ftp://ftp.symbios.com/pub/standards/io/

Information about IEEE 1394 implementations
http://developer.intel.com
http://www.microsoft.com/hwdev/busbios/

Open Host Controller Interface Specification
ftp://www.austin.ibm.com/pub/chrptech/1394ohci/

Windows NT DDK
MSDN Professional membership

Checklist for IEEE 1394

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

1. Controllers and devices support IEEE 1394-1995 standards
Required

2. Controllers comply with OpenHCI for IEEE 1394
Required

3. OpenHCI controllers and devices support advances defined in IEEE 1394A
Required

4. Host supports peak data rate of 400 Mb/s, minimum
Required

5. Design avoids excessive currents resulting from ground-fault potential among devices
Recommended

6. Device command protocols conform to standard device class interfaces
Required

7. Devices support peak data rate of 400 Mb/s, minimum
Recommended

8. Devices requiring support for high-bandwidth data transfer use IEEE 1394
Recommended

9. Plug and Play devices demonstrate interoperability with other devices
Required

10. Topology faults do not cause the bus to fail
Required

11. Removable media devices support media status notification
Required

12. Devices that can initiate peer-to-peer communications also support remote programming
Required

13. Device provides a configuration ROM for unique device identification
Required

14. Device configuration ROM implements general ROM format
Required

15. Bus information block implemented at a base address offset of 0404h
Required

16. Configuration ROM provides globally unique device ID
Required

17. Root directory is located at a fixed address following the bus information block
Required

18. Configuration ROM includes a unit directory for each independent device function
Required

19. Each unit directory provides a valid Unit_Spec_Id and Unit_Sw_Version
Required

20. Each unit directory provides a pointer to a unit-dependent directory
Required

21. Vendor and model leaves support textual descriptor leaf format
Required

22. Unit-dependent directory provides a pointer to the unit's CSRs
Required

23. Device provides three connector ports
Recommended

24. Device uses standard 6-pin IEEE 1394 connector
Required

25. Self-powered devices propagate the power bus through each connector
Required

26. Only single-port leaf-node devices use 4-pin connectors
Required

27. Device connectors exhibit common speed and power characteristics
Required

28. Standard 400-Mb/s rated IEEE 1394 cable provided with devices
Required

29. Devices power their PHY at all times
Required

30. Devices report power source and cable power consumption in Self_id packet
Required

31. Devices implement link power control
Required

32. Device requiring power increments in excess of Link_on implements unit-power CSRs
Required

33. Devices that source cable power must report this capability
Required

34. IEEE 1394-enabled PC sources cable power
Required

35. Power source supplies a minimum of 20 volts at 15 watts
Recommended

36. Devices notify the power manager of power change requests
Required

37. Devices and controllers comply with Cable Power Distribution specification
Required

38. Devices and controllers comply with IEEE 1394 power specification
Required

CHAPTER 9

PCI

9

This chapter presents the PC 98 requirements and recommendations for Peripheral Component Interconnect (PCI) host controllers and peripherals.

The PCI architecture has become the most common method used to extend PCs for add-on adapters. Windows and Windows NT use the basic PCI infrastructure to gain information about devices attached to the PCI bus. The ability of PCI to supply such information makes it an integral part of the Plug and Play architecture in Windows.

Specific requirements related to PCI are defined in the following chapters:

- Requirements for dual IDE adapters that use PCI are defined in the "IDE and ATAPI" chapter in Part 3 of this guide.
- Requirements for PCI-to-PC Card bridges are defined in the "PC Card" chapter in Part 3 of this guide.
- Requirements for graphics devices that use PCI are defined in the "Graphics Adapters" chapter in Part 4 of this guide.
- Requirements for audio implementations that use PCI are defined in the "Audio Components" chapter in Part 4 of this guide.

Contents

PCI Basic Requirements

This section summarizes the basic design requirements for PCI.

1. All components comply with PCI 2.1
Required

All cards, bridges, and devices that use PCI must be designed to meet the requirements defined in *PCI Local Bus Specification, Revision 2.1* (PCI 2.1).

Compliance with this requirement is demonstrated based on the compliance process of the PCI Special Interest Group (SIG).

2. System does not contain ghost cards
Required

A computer must not include any ghost cards, which are cards that do not decode the type 1/type 0 indicator. Such a card will appear on bus 0 as all the buses behind it that use the same IDSEL. Notice that it is acceptable, as defined in PCI 2.1, for a single-function card to decode the IDSEL and AD[1::0] pins and not decode AD[10::8] if the card does not have bit 7 set in the header type. This requirement also excludes, for example, devices that ignore some type 0 transaction bits and therefore appear at multiple device/function addresses.

A PCI card should be visible through hardware configuration access at only one bus/device/function coordinate.

3. System uses standard method to close BAR windows on nonsubtractive decode PCI bridges
Required

PCI-to-PCI bridges must comply with the *PCI to PCI Bridge Specification, Revision 1.0.* Setting the base address register (BAR) to its maximum value and the limit register to zeros should effectively close the I/O or memory window references in that bridge BAR.

4. System supports PCI docking through a bridge connector

Recommended

It is recommended that the system support docking through a bridge connector, with the actual bridge on the docking station, not on the mobile unit. The bridge can be positive or subtractive decoding. The bridge should create a new bus number so devices behind the bridge are not on the same bus number as other devices in the system.

After a warm dock, the BIOS should not configure the bridge or any other devices in the docking station. That is the responsibility of the operating system.

The PCI-to-ISA bridge should be placed on the docking station, not on the mobile unit. Mobile PCs typically do not have ISA expansion slots, and the ISA devices on the mobile PC can be controlled by the Plug and Play interface. For more information on requirements for docking station systems, see the “Basic PC 98” chapter in Part 2 of this guide.

Notice that implementing delayed transactions for PCI-to-PCI and PCI-to-ISA docking bridges is required in PCI 2.1 only when certain timing conditions are not met. For PC 98 design requirements, PCI 2.1 is interpreted to mean that delayed transactions are required only when “targets cannot complete the initial data phase within the requirements of this specification” (as stated in PCI 2.1). Delayed transactions are a hardware-related timing issue (and will provide a performance advantage), but are not related to operating system requirements.

5. PCI chip sets support Ultra DMA/33

Required

PCI chip sets must implement DMA as defined in SFF 8020i and must implement Ultra DMA/33 (also known as Ultra-ATA) as defined in the specification submitted by Quantum Corporation for inclusion in the ATA-4 specification.

Ultra DMA/33 is required to avoid the bottleneck created by the current 16.6-Mb/s limit on disk transfer. Ultra DMA/33 also provides error checking for improved robustness over previous IDE implementations.

PCI Controller Requirements

This section summarizes PCI controller requirements.

6. System-board bus complies with PCI 2.1
Required

The system-board bus hardware should comply with PCI 2.1. The bus design must fully implement all bus requirements on every expansion card connector.

7. Bus master privileges are supported for all connectors
Required

To ensure full Plug and Play functionality on a PCI bus with expansion cards, all PCI connectors on the system board must be able to allow any PCI expansion card to have bus master privileges.

8. ISA Write Data Port address is propagated to the ISA bus at power up
Required

If the system uses an ISA bus in conjunction with a PCI bridge, the Plug and Play ISA Write Data Port address must be propagated at all times through the bridge to all ISA buses that might contain external ISA Plug and Play cards. The address must be propagated at power up and system reset. This ensures that the system can identify, isolate, and configure external Plug and Play ISA cards plugged into the ISA bus during the boot process.

9. Functions in a multifunction PCI device do not share writable PCI Configuration Space bits
Required

The operating system treats each function of a multifunction PCI device as an independent device. As such, there can be no sharing between functions of writable PCI Configuration Space bits (such as the Command register).

Notice that the PC Card 16-bit Interface Legacy Mode BAR—offset 44h in the Type 2 PCI header—is the only exception to this requirement. This register must be shared between the two functions, just as they must share the same compatibility registers with the Exchangeable Card Architecture (ExCA) programming model, as defined in the *PCI to PCMCIA CardBus Bridge Register Description* (Yenta specification), by Intel.

For more information about design requirements for CardBus controllers, see the “PC Card” chapter in Part 3 of this guide.

Plug and Play for PCI Controllers and Peripherals

This section summarizes the Plug and Play requirements for PCI devices.

10. Devices use PCI 2.1 Configuration Space register for Plug and Play device ID

Required

The PCI 2.1 specification describes the Configuration Space register used by the system to identify and configure each device attached to the bus. The Configuration Space register is made up of a 256-byte field for each device and contains sufficient information for the system to identify the capabilities of the device. Configuration of the device is also controlled from this register.

The Configuration Space register is made up of a header region and a device-dependent region. Each Configuration Space register must have a 64-byte header at offset 0. All the device registers that the device circuit uses for initialization, configuration, and catastrophic error handling must fit in the space between byte 64 and byte 255.

All other registers that the device uses during normal operation must be located in normal I/O or memory space. Unimplemented registers or reads to reserved registers must complete normally and return zero (0). Writes to reserved registers must complete normally, and the data must be discarded.

11. Device IDs include PCI 2.1 Subsystem IDs

Required

The following diagram shows the two registers added to the Configuration Space header for PCI 2.1. Although these registers are only recommended in PCI 2.1, they are mandatory for PC 98. Support for these registers requires non-zero values to be populated for both the Subsystem ID and Subsystem Vendor ID.

New registers in Configuration Space header for PCI 2.1

31 16	15 0	
Subsystem ID	Subsystem Vendor ID	2Ch

These fields are necessary for the correct enumeration of a device. When the Subsystem ID fields are populated correctly for the adapter, Windows can differentiate between adapters based on the same PCI chip.

The Subsystem ID also allows Windows to load system miniports for system-board devices. Thus, Subsystem IDs are also a requirement on system-board devices. The exceptions to this requirement are PCI-to-PCI bridges and core chip sets.

Two methods can be used to implement a Subsystem Vendor ID:

- Load the value by hardware methods—for example, pin strappings at RST, an attached parallel or serial ROM, and so on.
- Program the Subsystem Vendor ID using BIOS. Two designs using the BIOS method meet PC 98 requirements:
  - Make a copy of the Subsystem Vendor ID in PCI user-defined space. Any writes to this location will change both the copy and the Subsystem Vendor ID field. Any writes to the Subsystem Vendor ID are discarded.
  - Make a write-enable bit in the PCI user-defined space. The BIOS can turn this bit on, change the Subsystem Vendor ID, and then turn it off.

For more information, see the article titled “IDs and Serial Numbers for Plug and Play” on the web site at http://www.microsoft.com/hwdev/busbios/.

Important: Multiple-monitor support allows display class devices to be initialized independently of the system initialization process. For this reason, system-board and add-on display devices cannot use the VGA BIOS POST routine to populate the Subsystem Vendor ID because the device’s POST code might not be executed until later in the process, after device enumeration occurs. For system-board devices, the system BIOS should populate the Subsystem Vendor ID at power on. Add-on display adapters should provide a method for populating the Subsystem Vendor ID at the point when power is applied and the device is initialized to the state that it is ready for POST.

12. Configuration Space is correctly populated

Required

Windows places extra constraints on a few configuration registers and has uncovered some problem usage of other registers. Microsoft provides a program named Pci.exe to help debug the use of the Configuration Space. This program is available on the Microsoft FTP server, as described in the “PCI References” section at the end of this chapter.

The following items are specific requirements for the Configuration Space:

- Populate the class code register (09h) for all devices.

  Follow the base class, sub-class, and programming interface values outlined in PCI 2.1.
- Devices must not fill BARs with random values.

  See PCI 2.1 for correct usage of these registers. Notice that BARs (10, 14, 18, 1C, 20, and 24h) should return zero if they are not used, indicating that no memory or I/O space is needed.

Also, for performance reasons, it is recommended that run-time registers for PCI devices should not be placed in the Configuration Space.

13. Interrupt routing supported using ACPI
Required

The system must provide interrupt routing information using a _PRT object, as defined in Section 6.2.3 of the *Advanced Configuration and Power Interface Specification, Revision 1.0* or higher.

14. BIOS does not configure I/O systems to share PCI interrupts
Recommended

This applies to boot devices configured by the BIOS on systems based on Intel Architecture processors. The operating system should configure all other devices. For systems that will run the Windows operating system, OEMs should design the BIOS so that it does not configure the I/O systems in the PC to share PCI interrupts for boot devices. An exception exists for legacy audio devices following the configuration guidelines outlined in the white paper titled *Implementing Legacy Audio Devices on the PCI Bus*, available on the web site at http://www.intel.com/pc-supp/platform/ac97/wp/leg_pci.htm.

Windows does not support sharing an IRQ between real-mode and protected-mode code within the I/O subsystem. An example of this is an NDIS 2.0 driver (real mode) and a SCSI miniport driver (protected mode) for two PCI devices that share the same IRQ. The problem is that the IRQ needs to be reflected to real mode for the NDIS 2.0 driver to work.

However, if the IRQ is reflected to real mode, the real-mode SCSI driver (which usually is not called because Windows takes over in protected mode) might touch the hardware, which would cause the SCSI miniport to be confused. Windows resolves this problem either by switching everything to protected mode or by falling back to real mode.

15. BIOS configures boot device IRQ and writes to the interrupt line register
Required

This requirement applies to boot devices configured by the BIOS on systems based on Intel Architecture processors. Windows should configure all other devices because, after an IRQ is assigned by the system BIOS, Windows cannot change the IRQ, even if necessary. If the BIOS assigns the IRQ and Windows needs it for another device, a sharing problem occurs.

The BIOS must configure the boot device IRQ to a PCI-based IRQ and must write the IRQ into the interrupt line register 3Ch, even if the BIOS does not enable the device. This way, the operating system can still enable the device with the known IRQ at configuration time, if possible.

16. Hot swapping for any PCI device uses ACPI-based methods
Required

Windows 98 and Windows NT 5.0 support dynamic enumeration, installation, and removal of PCI devices only if there is a supported hardware insert/remove notification mechanism.

The appropriate notification mechanism is supported as a bus standard for CardBus bus controllers. For other solutions, such as those required for docking stations or other devices, the hardware insert/remove notification mechanism must be implemented as defined in Section 5.6.3 of the ACPI 1.0 specification. To properly function with the native support in the operating system, developing industry standards, such as those referred to as PCI Hot Plug and Compact PCI, must use ACPI-based methods for supporting hardware insertion and removal as defined in the ACPI 1.0 specification.

Power Management for PCI Controllers and Peripherals

This section summarizes the specific PCI power management requirements.

17. All PCI components comply with PCI Bus Power Management Interface specification
Required

The PCI bus, any PCI-to-PCI bridges on the bus, and all devices on the PCI bus must comply with *PCI Bus Power Management Interface Specification, Revision 1.0* or higher. This specification defines the OnNow device power states (D0–D3) for PCI devices. It also covers the bus functionality expected in each power state and the mechanism for signaling wake-up events over the bus. For PC 98, PCI bus implementations must support all four device power states, either the B2 or B3 bus power state, and the standard wake-up mechanism.

This specification allows the operating system to individually power manage PCI devices to conserve power, and it allows PCI peripherals to wake up the system when the need arises.

Note: It is an acceptable alternative for embedded PCI devices (on the system board) to use ACPI for power-state and wake-up control instead of the new PCI definitions, as defined in Sections 3.3, 3.4, and 7.0 of the ACPI 1.0 specification.

For add-on adapters, including AGP cards, compliance with the PCI Bus Power Management specification is required, including the Configuration Space registers and the device state (D*x*) definitions.

PCI References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

Advanced Configuration and Power Interface Specification, Revision 1.0
http://www.teleport.com/~acpi/

"Efficient Use of PCI," Platform Architecture Labs, Intel Corporation
http://support.intel.com/oem_developer/chipsets/pci/general/pci001.htm

"IDs and Serial Numbers for Plug and Play" and other related articles
http://www.microsoft.com/hwdev/busbios/

Implementing Legacy Audio Devices on the PCI Bus
http://www.intel.com/pc-supp/platform/ac97/wp/leg_pci.htm

Microsoft testing tools, specifications, and information
E-mail: pciinfo@microsoft.com
http://www.microsoft.com/hwtest/
http://www.microsoft.com/hwdev/busbios/
ftp://ftp.microsoft.com/developr/drg/plug-and-play/pci/pci.exe

PCI Bus Power Management Interface Specification, Revision 1.0
PCI Local Bus Specification, Revision 2.1 (PCI 2.1)
PCI to PCI Bridge Specification, Revision1.0.
PCI SIG
Phone: (800) 433-5177
http://www.pcisig.com/

PCI to PCMCIA CardBus Bridge Register Description (Yenta specification)
PCMCIA
2635 North First Street, Suite 209
San Jose, CA 95134 USA
Phone: (408) 433-2273
Fax: (408) 433-9558
E-mail: office@pcmcia.org
http://www.pc-card.com/

Checklist for PCI

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

1. All components comply with PCI 2.1
Required

2. System does not contain ghost cards
Required

3. System uses standard method to close BAR windows on nonsubtractive decode PCI bridges
Required

4. System supports PCI docking through a bridge connector
Recommended

5. PCI chip sets support Ultra DMA/33
Required

6. System-board bus complies with PCI 2.1
Required

7. Bus master privileges are supported for all connectors
Required

8. ISA Write Data Port address is propagated to the ISA bus at power up
Required

9. Functions in a multifunction PCI device do not share writable PCI Configuration Space bits
Required

10. Devices use PCI 2.1 Configuration Space register for Plug and Play device ID
Required

11. Device IDs include PCI 2.1 Subsystem IDs
Required

12. Configuration Space is correctly populated
Required

13. Interrupt routing supported using ACPI
Required

14. BIOS does not configure I/O systems to share PCI interrupts
Recommended

15. BIOS configures boot device IRQ and writes to the interrupt line register
Required

16. Hot swapping for any PCI device uses ACPI-based methods
Required

17. All PCI components comply with PCI Bus Power Management Interface specification
Required

C H A P T E R   1 0

IDE and ATAPI

This chapter presents the PC 98 requirements and recommendations for ATA (AT Attachment), ATAPI (ATA Packet Interface) controllers and peripherals. ATA is more commonly known as IDE (Integrated Device Electronics).

The IDE interface is one of the most widely used in the PC world. Originally intended only for hard drives, IDE support is being extended to additional device types and performance features.

The use of IDE in a PC 98 system is optional. If IDE is used, however, all components must comply with the requirements defined in this chapter.

Contents

IDE Controller Requirements

This section summarizes the specifications and standards for Windows-compatible IDE controllers.

1. Controller complies with ATA-2 specification
Required

Recommended: ATA-3 compliance.

All IDE adapters (and peripherals) must meet the hardware and software design requirements listed in the current version of the *AT Attachment 2* specification.

2. Bootable IDE controller supports El Torito No Emulation mode
Required

A bootable IDE storage controller must support the No Emulation mode defined in *El Torito—Bootable CD-ROM Format Specification, Version 1.0,* by IBM and Phoenix, or an equivalent method that supports the Windows NT CD-ROM installation process.

3. System BIOS and option ROMs support Int 13h Extensions
Required

The Int 13h Extensions ensure correct support for high-capacity drives, consistent drive-letter mapping between real mode and protected mode, and other capabilities for both Windows and Windows NT. Support for the fixed-disk access subset of Int 13h Extensions must be provided in the system BIOS and in any option ROMs for storage devices that include BIOS support.

The Int 13h Extensions are defined in the Windows NT 5.0 DDK and in the "Layered Block Device Drivers" section of the Windows 98 DDK.

4. Controller and peripherals support media status notification
Required

For CD-ROM and DVD-ROM, manufacturers must comply with all provisions defined in the Media Status Event Notification subsection of SFF 8090 (Mt. Fuji specification). This specification is available from the SFF Committee and from ftp://fission.dt.wdc.com/pub/standards/SFF/specs/.

For ATAPI floppy drives, manufacturers must implement media status notification as defined in SFF 8070.

For other ATA and ATAPI devices, *Media Status Notification Support Specification, Version 1.03* or higher, by Microsoft Corporation, defines the protocol to use for communicating about the current state of removable media. This specification is available at http://www.microsoft.com/hwdev/specs/.

For other ATAPI devices such as tape drives, media status notification is not required, but if it is implemented, the support must comply with SFF 8090.

Important: For CD-ROM and DVD-ROM devices, do not use *Media Status Notification Support Specification, Version 1.03* or earlier, as the guideline for implementing status notification on optical storage devices. This specification does not apply to optical storage devices because it does not contain packet-based support.

5. Dual IDE adapters use single FIFO with asynchronous access or dual FIFOs and channels

Required

PCI dual IDE adapters must be designed so that either channel might be used at any time; that is, the operating system does not have to serialize access between the primary and secondary channel at any time. This means either that the two channels are totally independent or that anything shared, such as a programmed I/O (PIO) read pre-fetch buffer, is protected by a hardware arbitrator.

A design implementing a single first in/first out (FIFO) that uses a hardware solution to synchronize access to both channels meets this requirement if the design does not require that a request on one channel be completed before another can be started.

Section 5.0 of the *Compaq, Intel, Phoenix BIOS Boot Specification* defines the implementation for dual asynchronous channels.

Dual-channel controllers that require special software to serialize channel I/O for a single prefetch FIFO do not meet these requirements. Such designs require serial access to one of four devices, defeating the primary advantage of asynchronous dual-channel controllers. Furthermore, such devices are non-standard and require custom driver support.

The introduction of non-standard IDE hardware is strongly discouraged because it negatively impacts traditional compatibility of the IDE interface. Notice, however, that dual-channel controllers which do not require special software to serialize channel I/O do meet these requirements.

6. System BIOS and devices support LBA

Required

To enable support for IDE disk drives that are larger than 528 MB, the system BIOS must use a logical block addressing (LBA) scheme that is compatible with the BIOS/CMOS and IDE register set constraints. The system BIOS must also be able to enable and disable block mode, and must be able to disable 32-bit mode.

Although ATAPI was defined to be transparent to the BIOS, the BIOS must recognize the presence of ATAPI devices using the signature defined in SFF 8020i. In some cases, without such support, the BIOS might fail to configure the adapter if it does not see a device.

7. Controller and peripherals support PCI IDE bus mastering

Required

The programming register set for PCI IDE bus master DMA is defined in SFF 8038i. IDE drives must comply with SFF 8038i to ensure fully featured hardware and Windows-compatible device driver support.

With ATAPI CD-ROM, PIO demands placed on the system CPU can have a negative impact on performance and application processing, especially for multimedia. Bus master DMA IDE adapters, which leverage local bus data rates, can provide higher data rates and the ability to offload the system CPU from I/O transfers.

Other factors that encourage the adoption of bus master DMA include the increased disk media transfer rates, plus demands made by multitasking operating systems and multichannel/multidevice IDE configurations.

Controllers and peripherals must also support Ultra DMA/33 (also known as Ultra-ATA) as defined in the "ATAPI Peripheral General Requirements" section later in this chapter.

8. Controller and peripheral connections include Pin 1 cable designation with keyed and shrouded connectors

Required

Pin 1 orientation must be designated by one edge of the keyed ribbon cable and also on the keyed connector of the IDE or ATAPI controller and peripheral device. Designation of the keyed connector must be clearly indicated on or near the connector.

ATAPI Peripheral General Requirements

This section defines the requirements for all ATAPI devices. Specific requirements for IDE floppy drives, hard drives, CD-ROM, and DVD devices are defined in the "Storage and Related Peripherals" chapter in Part 4 of this guide.

9. Peripherals comply with SFF 8020i, Version 2.5 or higher

Required

This specification defines standard hardware and software design guidelines for ATAPI devices. See also the "System BIOS and option ROMs support Int 13h Extensions" requirement earlier in this chapter.

10. BIOS enumeration of all ATAPI devices complies with SFF 8020i, Version 2.5 or higher

Required

ATAPI specification SFF 8020i, Version 2.5 or higher, defines the enumeration process for all ATAPI devices.

11. Devices support ATAPI RESET command

Recommended

This item ensures that the ATAPI RESET command is processed by the peripheral, even if the firmware state cannot be determined. Reset the controller by going into a power-on state (requests cleared, signature present), except leave any nondefault mode values in their current state and leave the DRV bit unchanged. For more information, see Section 6.2 of SFF 8020i, Version 2.5 or higher.

12. IDE/ATAPI controllers and devices support Ultra DMA/33

Required

Ultra DMA/33 is required to avoid the bottleneck created by the current 16.6-Mb/s limit on disk transfer. Ultra DMA/33 also provides error checking for improved robustness over previous IDE implementations. This is a requirement for all IDE/ATAPI controllers and devices.

PCI chip sets must implement DMA as defined in SFF 8020i and must implement Ultra DMA/33 as defined in the specification submitted by Quantum Corporation for inclusion in the ATA-4 specification (proposed as ATA-4 1153 DR11).

Plug and Play for IDE Controllers and Peripherals

This section summarizes the Plug and Play requirements for IDE controllers and peripherals.

13. Operating system recognizes the boot drive in a multiple-drive system
Required

The implementation of boot-drive determination in multiple-drive systems is defined in Section 5.0 of the *Compaq, Intel, Phoenix BIOS Boot Specification*. This is the format that both Windows and Windows NT operating systems use for determining the boot drive as new bootable devices are introduced for PCs. The system designer can use an equivalent method for boot-drive determination, but the method must ensure that the Windows and Windows NT operating systems recognize the boot drive.

14. Each device has a Plug and Play device ID
Required

For a system-board device, there must be a Plug and Play device-specific ID.

Each IDE controller or peripheral device must provide device IDs in the manner required for the bus it uses, as defined in the related chapter for the specific bus in Part 3 of this guide.

For example, an add-on PCI IDE device must comply with PCI 2.1 and also must provide a Subsystem ID and Subsystem Vendor ID as defined in the "PCI" chapter in Part 3 of this guide. PCI IDE controllers integrated into core logic on the system board do not have to provide Subsystem IDs and Subsystem Vendor IDs, but must meet other PCI 2.1 requirements.

15. Dynamic resource configuration supported for all devices
Required

All devices must be capable of being automatically disabled by the system. Also, disabling the device must result in freeing all its resources for use by other devices.

Changing or adding a controller to the system must not require changing jumpers or switches on either the controller or the system board.

16. Resource configuration meets bus requirements
Required

Plug and Play resource-configuration requirements are defined by the bus used by the IDE/ATAPI controllers and peripheral devices, as defined in the related chapter for the specific bus in Part 3 of this guide.

17. ISA address ranges 3F7h and 377h are not claimed by IDE controllers
Required

Although IDE controllers might use these addresses, 3F7h and 377h also contain registers used by the FDC. To prevent resource conflicts, these addresses must not be claimed as device-register resources.

Power Management for IDE Devices

This section summarizes the specific IDE power management requirements. Power management requirements for peripherals that use IDE are defined in the related device-class chapters in Part 4 of this guide.

18. Device supports ATA STANDBY command
Required

The IDE drive must implement the ATA STANDBY command according to the ATA standard to ensure that the drives are able to spin up properly after a STANDBY command. This command is defined in the ATA-2 specification and in SFF 8020i.

It is recommended that the hard disk drive spin-up and be able to complete a Read operation within 6 seconds of applying power and within 5 seconds of leaving ATA STANDBY mode and transitioning to ATA ACTIVE, as specified in the *Storage Device Class Power Management Reference Specification, Version 1.0* or higher.

19. Bus and device meet PC 98 power management requirements
Required

The IDE channel must comply with the *Storage Device Class Power Management Reference Specification, Version 1.0* or higher. Additional power management requirements are specified based on industry-defined standards for the bus used by the controller (such as PCI) and for the device. For more information, see the related chapter for the specific bus in Part 3 of this guide.

IDE and ATAPI References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

AT Attachment 2 [X3T9.2 948D] and other ATA specifications
ATA Packet Interface for CD-ROM, SFF 8020i
Other SFF Committee publications
ftp://fission.dt.wdc.com/pub/standards/SFF/specs/

Compaq, Intel, Phoenix BIOS Boot Specification, Version 1.01
http://www.ptltd.com/techs/specs.html

El Torito—Bootable CD-ROM Format Specification, Version 1.0
Compaq, Intel, Phoenix BIOS Boot Specification, Version 1.01
http://www.ptltd.com/techs/specs.html

IDE and SCSI specifications
SFF Committee publications
FaxAccess: (408) 741-1600 (fax-back)
Fax: (408) 867-2115

Global Engineering Documents
Phone: (800) 854-7179 (US)
(613) 237-4250 (Canada)
(303) 792-2181 (Outside North America)
Fax: (303) 397-2740
ftp://ftp.symbios.com/pub/standards/io/

Media Status Notification Support Specification, Version 1.03
SMART IOCTL API Specification, Version 1.1
http://www.microsoft.com/hwdev/specs/

PCI Local Bus Specification, Revision 2.1 (PCI 2.1)
PCI SIG
Phone: (800) 433-5177
http://www.pcisig.com

Storage Device Class Power Management Reference Specification, Version 1.0
http://www.microsoft.com/hwdev/onnow.htm

Ultra DMA/33 specification
http://www.quantum.com

Windows DDK and Windows NT DDK
MSDN Professional membership

Checklist for IDE and ATAPI

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

1. Controller complies with ATA-2 specification
Required

2. Bootable IDE controller supports El Torito No Emulation mode
Required

3. System BIOS and option ROMs support Int 13h Extensions
Required

4. Controller and peripherals support media status notification
Required

5. Dual IDE adapters use single FIFO with asynchronous access or dual FIFOs and channels
Required

6. System BIOS and devices support LBA
Required

7. Controller and peripherals support PCI IDE bus mastering
Required

8. Controller and peripheral connections include Pin 1 cable designation with keyed and shrouded connectors
Required

9. Peripherals comply with SFF 8020i, Version 2.5 or higher
Required

10. BIOS enumeration of all ATAPI devices complies with SFF 8020i, Version 2.5 or higher
Required

11. Devices support ATAPI RESET command
Recommended

12. IDE/ATAPI controllers and devices support Ultra DMA/33
Required

13. Operating system recognizes the boot drive in a multiple-drive system
Required

14. Each device has a Plug and Play device ID
Required

15. Dynamic resource configuration supported for all devices
Required

16. Resource configuration meets bus requirements
Required

17. ISA address ranges 3F7h and 377h are not claimed by IDE controllers
Required

18. Device supports ATA STANDBY command
Required

19. Bus and device meet PC 98 power management requirements
Required

CHAPTER 11

SCSI

This chapter presents the PC 98 requirements and recommendations for the small computer system interface (SCSI). The use of SCSI in a PC 98 system is optional, but if SCSI is used, all components must comply with the requirements defined in this chapter.

SCSI is a flexible I/O bus that is used in the design of a wide variety of peripherals, including disk drives, CD-ROM drives, tape drives, scanners, and magneto-optical drives. The SCSI host adapter is the circuitry that serves as an interface between the system and one or more SCSI peripherals. A host adapter can be a card that plugs into the system's expansion bus, such as a PCI card, or it can be designed directly into the system board.

Contents

SCSI Host Adapter Requirements

This section summarizes class specifications and standards for SCSI host adapters.

1. Host controller supports bus mastering
Required

The host controller must support bus mastering.

2. Bootable SCSI controller supports El Torito No Emulation mode
Required

A bootable SCSI storage controller must support the No Emulation mode defined in *El Torito—Bootable CD-ROM Format Specification, Version 1.0,* by IBM and Phoenix, or an equivalent method that supports the Windows NT CD-ROM installation process.

3. Option ROMs support Int 13h Extensions
Required

The Int 13h Extensions ensure correct support for high-capacity drives, consistent drive-letter mapping between real and protected modes, and other capabilities for both Windows and Windows NT. Support for the fixed-disk access subset of Int 13h Extensions must be provided in the system BIOS and in any option ROMs for storage devices that include BIOS support. The Int 13h Extensions are defined in the Windows NT 5.0 DDK and in the “Layered Block Device Drivers” section of the Windows 98 DDK.

4. Option ROMs support virtual DMA services
Required

Plug and Play SCSI host adapters must support virtual DMA services in the host-adapter option ROM and must support bus mastering. Virtual DMA supports scatter/gather capabilities, solving the problem of mapping linear addresses (segment:offset) into physical addresses.

5. Bus type is clearly indicated on connectors for all adapters, peripherals, cables, and terminators
Required

Connectors for each SCSI adapter, peripheral, cable, and terminator must be clearly labeled to indicate the bus type. All external SCSI connectors must display the appropriate SCSI icon defined in *Small Computer Interface* (SCSI-3) *Parallel Interface* (SPI) specification, Annex F, and must display any clarifying abbreviations or acronyms.

6. Differential devices support DIFFSENS as defined in SCSI-3 specification
Required
Without DIFFSENS, the differential bus drivers and/or a single-ended device will burn up if a single-ended device is put on a differential bus.

The specification for DIFFSENS is defined in Section 5.4.2 of the SCSI-3 specification.

7. Automatic termination circuit meets SCSI-3 specification
Required
SCSI add-on adapters and on-board controllers must use automatic termination, which allows a user to add external devices without removing the PC case. Terminators used in the SCSI host adapter must be regulated terminators, also known as active, SCSI-3 SPI, SCSI-2 alternative-2, or Boulay terminators.

8. SCSI terminator built onto internal cables meets SCSI-3 specification
Required
For SCSI subsystem configurations, internal cables must be preconfigured with active termination at one end of the cable.

9. Terminator power is supplied to the SCSI bus with over-current protection
Required
For system-board implementations using PCI or another expansion bus, the host adapter must supply terminator power (TERMPWR) to the SCSI bus. All terminators on the host adapter, as well as those on the internal and external SCSI bus, must be powered from the TERMPWR lines on the SCSI bus.

Devices that provide TERMPWR must also provide some means of limiting the current, through use of a self-setting device. For example, a positive-temperature coefficient device or circuit breaker can be designed into the circuit. These devices open during an over-current condition and close after the condition ends.

This item is a recommendation for battery-powered systems that implement the SCSI host adapter as a PC Card device, because of battery power-consumption issues.

10. External connector meets SCSI-2 or higher specification
Required
If an external connector is implemented, it must be a high-density connector and must meet the requirements defined in the SCSI-2 or higher specification.

11. Internal terminator is as close as possible to the last peripheral on the cable

Recommended

The internal terminator should be as physically close as possible to the last peripheral on the cable. There should be some means, such as written instructions on the cable, to ensure that the user always connects internal peripherals starting with the plug closest to the terminator.

SCSI Peripheral Requirements

This section summarizes requirements related to specifications and standards for SCSI peripherals.

12. SCSI bus parity signal meets SCSI-2 specification

Required

All SCSI peripherals must implement the SCSI bus parity signal defined in the SCSI-2 specification.

13. Cables meet SCSI-3 Clause 6 requirements

Required

Clause 6 of the SCSI-3 specification defines the various characteristics of cables for SCSI devices.

14. User cannot incorrectly plug in cables

Required

For an internal configuration, the internal SCSI bus cable must be plugged into shrouded and keyed connectors on the host adapter and devices. This ensures that the cable is properly positioned. Pin 1 orientation must be designed on one edge of the ribbon cable and also on the keyed connector of the SCSI peripheral device.

For an external configuration, the SCSI connector must not use the same connector type as any non-SCSI connector on the system.

15. External devices use automatic termination or an accessible on-board termination switch

Required

The recommended implementation for an external SCSI peripheral device is to provide automatic termination. In the absence of automatic termination, an external pluggable terminator must be connected to the last open device connector on a bus. If a mechanical means is provided for setting termination, the switch must be accessible to the user without opening the PC case.

16. Shielded device connector meets SCSI-2 or higher specification

Required

Device connectors must meet the specifications defined in the SCSI-2 or higher specification.

17. Removable media includes media status notification support

Recommended

A specification has not yet been completed for implementing media event status notification. However, the projected specification will be similar to the Media Event Status Notification subsection of SFF 8090 (Mt. Fuji specification). This specification is available at ftp://fission.dt.wdc.com/pub/standards/SFF/specs/.

When a completed specification is available, support for media status notification will become a requirement.

Plug and Play for SCSI Host Adapters and Peripherals

This section summarizes the Plug and Play requirements for SCSI devices.

18. Each device has a Plug and Play device ID

Required

For a system-board device, there must be a Plug and Play device-specific ID.

Each SCSI controller or peripheral device must provide device IDs as defined in the *Plug and Play SCSI Specification, Version 1.0*, and in the specification for the bus it uses as defined in the related chapter in Part 3 of this guide. For example, a PCI device must comply with PCI 2.1 and also must provide a Subsystem ID and Subsystem Vendor ID as defined in the “PCI” chapter in Part 3 of this guide.

PCI controllers integrated into core logic on the system board do not have to provide Subsystem IDs and Subsystem Vendor IDs, but must meet other PCI 2.1 requirements.

19. Automatic resource assignment and dynamic disable capabilities supported for all devices

Required

For SCSI on-board controllers and add-on adapters, the system must be capable as necessary of automatically assigning, disabling, and relocating the resources used by the device. Changing this device or adding it to the system must not require changing jumpers or switches on either the adapter or the system board. In the event of an irreconcilable conflict with other devices, the operating system must be able to disable the device in order to prevent the system from stalling.

20. SCSI controllers provide multi-initiator support

Recommended

Multi-initiator support allows two SCSI controllers—each installed in a separate computer system—to coexist on a shared SCSI bus with a set of shared devices. If this capability is supported, the SCSI IDs must be changeable from the default SCSI controller ID of 7, and the boot-time SCSI bus reset operation must be able to be disabled on each controller attached to a shared bus.

Power Management for SCSI Devices

This section summarizes the specific power management requirements for the SCSI bus class. Power management requirements for other device classes are defined in Part 4 of this guide.

21. Bus and device meet PC 98 power requirements

Required

Additional power management requirements are specified based on industry standards for the bus used by the controller and for the device. For more information, see the related chapter for the specific bus class in Part 3 in this guide. See also Part 4 of this guide for the related device class requirements based on compliance with the specific device class power management reference specification.

22. Hardware supports the STOP/START UNIT command as defined in the SCSI-2 specification

Required

The hardware in SCSI peripherals must be able to fully recover from a software-initiated spin down without rebooting the system or cycling power. To properly support power management on SCSI drives and to ensure that the operating system responds to appropriate driver calls, be sure to correctly implement the STOP/START UNIT command as defined in the SCSI-2 specification.

23. STOP/START UNIT command can be used to decrease power consumption

Recommended

Wherever appropriate—for example, for storage disks—the STOP UNIT command can be used to decrease the power consumption of the base platform. In order for any form of power management to work on SCSI, the device should be capable of supporting many tens of thousands of START/STOP UNIT commands over the life of a device.

SCSI References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

Device class power management reference specifications
http://www.microsoft.com/hwdev/onnow.htm

El Torito—Bootable CD-ROM Format Specification, Version 1.0
Compaq, Intel, Phoenix BIOS Boot Specification, Version 1.01
http://www.ptltd.com/techs/specs.html

PCI Local Bus Specification, Revision 2.1 (PCI 2.1)
PCI SIG
Phone: (800) 433-5177
http://www.pcisig.com

Plug and Play SCSI Specification, Version 1.0
http://www.microsoft.com/hwdev/specs/

Small Computer Interface (SCSI-2) [X3T9.2-375R] specification
Small Computer Interface (SCSI-3) *Parallel Interface* (SPI) [X3T9.2/91-10] specification
SFF Committee publications
FaxAccess: (408) 741-1600 (fax-back)
Fax: (408) 867-2115

Global Engineering Documents
Phone: (800) 854-7179 (US)
(613) 237-4250 (Canada)
(303) 792-2181 (Outside North America)
Fax: (303) 397-2740
ftp://ftp.symbios.com/pub/standards/io/

Windows and Windows NT DDK
MSDN Professional membership

Checklist for SCSI

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

1. Host controller supports bus mastering
Required

2. Option ROMs support Int 13h Extensions
Required

3. Option ROMs support virtual DMA services
Required

4. Bus type is clearly indicated on connectors for all adapters, peripherals, and terminators
Required

5. Differential devices support DIFFSENS as defined in SCSI-3
Required

6. Automatic termination circuit meets SCSI-3 specification
Required

7. SCSI terminator built onto internal cables meets SCSI-3 specification
Required

8. Terminator power is supplied to the SCSI bus, with over-current protection
Required

9. High-density external connector meets SCSI-2 specification
Required

10. Internal terminator is close as possible to the last peripheral on the cable
Recommended

11. SCSI bus parity signal meets SCSI-2 specification
Required

12. Cables meet SCSI-3 Clause 6 requirements
Required

13. User cannot incorrectly plug in cables for internal connections
Required

14. Internal SCSI peripherals do not terminate the SCSI bus
Recommended

15. External connectors use automatic termination or an accessible on-board termination switch
Required

16. High-density, shielded device connector meets SCSI-2 specification
Recommended

17. Removable media includes media status notification support
Recommended

18. All components comply with Plug and Play SCSI specifications
Required

19. Each device has a Plug and Play device identifier
Required

20. Automatic resource assignment and dynamic disable capabilities are supported for all devices
Required

21. Bus and device meet PC 98 power requirements
Required

22. Hardware supports the STOP/START UNIT command as defined in SCSI-2
Required

23. STOP/START UNIT command can be used to decrease power consumption
Recommended

C H A P T E R 1 2

PC Card

This chapter presents PC 98 requirements and recommendations for PC Card. This includes PC Card 16 (previously referred to as PCMCIA cards), CardBus cards, and PC Card socket controllers (both PC Card 16-only controllers and CardBus controllers).

Windows 98 supports PC Card 16 I/O cards. Memory PC Card 16 cards are supported only as legacy devices. For any PC Card device to work effectively with Windows, the manufacturer must implement a minimum set of tuples documented in the PC Card standard. Windows uses these tuples to identify and configure any PC Card 16 card.

Contents

PC Card Basic Requirements

This section summarizes the basic PC 98 requirements for PC Card.

1. All devices comply with the PC Card standards
Required

Designs for PC Card socket controllers and cards must all be based on the PC Card standards.

The *February 1995 PC Card Standard* added requirements for minimum card information structure (CIS), 3.3-volt cards, multifunction cards, and cards that use DMA. It also introduced requirements for CardBus (32-bit) cards. The *March 1997 PC Card Standard* incorporated all corrections, changes, and adopted proposals as of that date.

All PC Card devices must comply with these standards for PC 98.

2. System and ZV-compatible PC Card 16 cards comply with ZV standard definitions
Required

The PC Card standards define the requirements for ZV cards and system support.

PC Card Socket Controller Requirements

This section summarizes PC 98 requirements and standards for socket controllers.

3. Controller supports industry-standard ExCA register set
Required

The built-in software supporting PC Card 16 cards in Windows includes drivers for the industry-standard Exchangeable Card Architecture-compatible (ExCA-compatible) socket controllers. To be compatible with these drivers, socket-controller implementations must support the industry-standard ExCA base register set.

Notice that some controllers do not fully implement the register set and therefore are incompatible. Also, some controllers implement extended registers or enhancements. The built-in Windows drivers do not exploit these features, even though the controller might be compatible.

4. System maintains mapping of IRQ Routing Register bits to system interrupt vectors

Required

The system design must maintain the mapping of the PC Card controller's IRQ Routing Register bits to system interrupt vectors. This means that when an interrupt is programmed in the controller to occur on the IRQ*x* pin, the system's IRQ routing causes the interrupt controller to generate the interrupt vector for IRQ*x* and no other IRQ.

5. IRQ connections can be determined by using the 0805 register

Required

Windows uses the 0805 register on both PC Card 16-only controllers and on CardBus controllers to determine which ISA IRQs are connected to the controller. This register must engage (that is, drive low) the corresponding ISA IRQ when programmed with a value. It must disengage the IRQ (that is, float high) when programmed at zero (0).

6. CardBus controllers support both ISA and PCI interrupts

Required

PC Card software dynamically configures the bridge to use ISA interrupts for PC Card 16 cards and to use PCI interrupts for CardBus cards. As defined in the "IRQ connections can be determined by using the 0805 register" and "System maintains mapping of IRQ Routing Register bits to system interrupt vectors" requirements earlier in this section, CardBus controllers must maintain mapping of IRQ routing. Also, notice that systems implementing CardBus controllers must fully support PCI 2.1 as well as additional PCI requirements for IRQ routing as described in the "PCI" chapter in Part 3 of this guide.

7. System supports industry-standard definition for CardBus bridges

Required

If the system supports CardBus, it must support the definition in *PCI to PCMCIA CardBus Bridge Register Description* (Yenta specification) for CardBus controllers (PCI-to-CardBus bridges). This definition includes a common PCI Configuration Space header assigned the Header Type field value of 82h.

Although this requirement is not yet incorporated into the PCI standard, Windows supports it. Any controller features that are not part of the Yenta specification will not be used in standard drivers. The BIOS is responsible for any hardware initialization or setup required to make the controller comply with the Yenta specification or other requirements listed in this chapter.

Because CardBus host controllers are PCI bus bridges, they will be supported (enumerated and configured) by the PCI software in Windows in the same manner as other PCI bus bridges. For more information, see the "PCI" chapter in Part 3 of this guide.

8. BIOS initializes CardBus controller in 82365-compatible mode and supports backward compatibility

Recommended

CardBus controllers are enumerated and configured in Windows in the same way as other PCI bus bridges. The PCI bus bridge support in Windows 98 is based on new requirements for PCI interrupt routing and bridge-window configuration. Because of this, full compliance with the latest PCI specifications is a requirement for CardBus support.

There are steps the BIOS can take to achieve backward compatibility with Windows. Specifically, the BIOS can initialize the CardBus controller in Intel 82365-compatible mode and report it as device “PNP0E03, Intel 82365-compatible CardBus controller.” The requirements are as follows for BIOS POST time (CardBus controller ConfigSpace initialization):

- Command register (that is, offset 0x04) set to 0x07 (that is, IOSpaceEnable, MemSpaceEnable, BusMasterEnable).
- RegisterBaseAddress (that is, offset 0x10) set to 0. If support for other environments is needed (such as Windows 3.1 or MS-DOS®), some other value may be set.
- All memory and I/O windows (that is, offset 0x1c–0x38) set to 0.
- Interrupt Line register (that is, offset 0x3c) set to 0xff (no IRQ assigned). If support for other environments is needed (such as Windows 3.1 or MS-DOS), an assigned IRQ line can be set. Notice, however, that this register must be set to 0xff at the time that the device is disabled by the operating system, and then set into CardBus mode. More information about BIOS enumeration is presented later in this requirement.
- Other controller-specific initialization as required to put the controller in legacy mode.

This puts the CardBus controller into legacy mode where the Windows Socket Services driver can access it as an Intel PC Card I/O card–compatible (PCIC-compatible) controller at an I/O address (for example, 0x3e0).

Notice that the BIOS must be at least PCI 2.1-compliant and must support the $PIR Interrupt Routing Table. The $PIR table must return the necessary PCI IRQ routing information, including the routing information for the CardBus controller. In general, if the CardBus controller is on the system board, there must be a slot routing entry for it in the table. If the CardBus controller is a PCI add-on card, there must be routing information entries for each PCI slot in the system.

During Plug and Play BIOS enumeration, the BIOS should report the CardBus controller as *pnp0e03 with a compatible ID of *pnp0e00 and the I/O resource of two ports (for example, 0x3e0–0x3e1).

For more information, see the white paper on CardBus host controllers and Windows compatibility at http://www.microsoft.com/hwdev/busbios/.

9. CardBus controllers do not share writable PCI Configuration Space bits
Required

CardBus controllers are multifunction PCI devices, and Windows treats each function as an independent device. As such, there can be no sharing between functions of writable PCI Configuration Space bits (such as the Command register).

Notice that the PC Card 16-bit interface legacy-mode BAR (offset 44h in the Type 2 PCI header) is the only exception to this requirement. This BAR must be shared between the two functions, since they must share the same BARs in order to be compatible with the ExCA programming model.

10. Each PC Card 16 memory window in CardBus controller has it own page register
Required

For complete flexibility and support of typical configurations, CardBus controllers must support the independent location of R2 memory windows anywhere in the full system address space as recommended in the Yenta specification.

Controllers that share a single page register among all PC Card 16 memory windows require that all PC Card 16 memory windows must be located within the same 16-MB block. Often, this is not possible with typical (16 MB) DRAM and bridge (positive-decode) configurations. The result is disabled cards.

Plug and Play Design for PC Card 16 Cards

This section summarizes the Plug and Play requirements for PC Card 16 cards.

The Windows operating system determines what type of card is plugged into the PC Card socket by examining the tuples on the card. For Plug and Play functionality, PC Card 16 I/O cards must support a set of required information and configuration tuples. The PCMCIA bus enumerator uses these tuples to identify the card, load the correct device driver, and indicate all possible configurations to the Plug and Play configuration manager. The operating system then dynamically assigns a valid configuration based on this information.

11. Card supports required I/O card tuples

Required

The following items must be implemented for any PC Card 16 I/O card that connects to a PC 98 system:

- The PC Card 16 card must contain the device information tuple (CISTPL_DEVICE, 01h), the Level 1 (L1) version/product information tuple (CISTPL_VERS_1, 15h), the configuration tuple (CISTPL_CONFIG, 1Ah), and the configuration table entry tuple (CISTPL_CFTABLE_ENTRY, 1Bh).
- The L1 version/product information tuple must contain the product name and manufacturer name in the product information string (TPLLV1_INFO, byte 4).
- The product name and manufacturer name in the L1 version/product information tuple must be composed only of ASCII characters greater than ASCII 20h and less than ASCII 7Fh.

Windows uses the information contained in the required and recommended tuples to create a unique device ID for the card and to assimilate configuration information for the device. Windows uses the device configuration tuples to determine the general characteristics of the card.

Required I/O Card Tuples

Tuple ID	Tuple code	Description and comments
01h	CISTPL_DEVICE	Device information (common memory). For non-memory cards, this tuple must be present, but the device type will be NULL.
15h	CISTPL_VERS_1	L1 version/product information strings: Product information Product name Product number Other manufacturer information
1Ah	CISTPL_CONF	Configuration. Indicates the location of configuration registers and registers present.
1Bh	CISTPL_CE	Configuration table entry. Appropriate configuration requirements for I/O space, interrupts, memory, and so on should be specified.
20h	CISTPL_MANFID	Manufacturer ID. Card manufacturer ID code. Defines manufacturer for this card.
21h	CISTPL_FUNCID	Function ID. Provides function information about the card. Also includes system initialization information.

The device information tuple provides information about the memory devices used in the card's common memory space. The device type, size, and speed are used to configure the socket for efficient access to the card. This tuple must be present on PC Card 16 I/O cards, but the device type must be NULL.

The L1 version/product information tuple contains human-readable information about the product and its manufacturer. This information is intended to be displayed to the user where necessary. Windows uses the information contained in the product information string and product name string to construct the device ID for that card. It also scans through the tuple, starting at the very beginning and continuing to the end of the product name string.

The information gathered from the L1 version/product information tuple is used to construct the unique device ID. Because the optional third and fourth strings in the tuple are not used in the unique ID, devices that require unique numbers on each card can use these strings to store that information.

The configuration tuple tells the software where to locate the configuration registers that program the card's configuration, as well as which registers are present on the card.

Each configuration table entry tuple completely describes one valid configuration in which the card can operate. Each entry describes power, timing, I/O space, interrupt, and memory space requirements for the given configuration. Configuration software selects one of these configurations for the card based on the resources currently available in the system.

The manufacturer ID tuple (CISTPL_MANFID, 20h) and the function ID tuple (CISTPL_FUNCID, 21h) add extra flexibility to a PC Card that connects to the PC:

- The manufacturer ID tuple provides unique information about the card manufacturer. This code is registered with PCMCIA. Windows uses the manufacturer ID tuple as one source for creating a 16-bit CRC used in the construction of the device ID.
- The function ID tuple provides information about the class of device or what function the card provides (for example, memory, modem, disk, and so on). This information helps the software perform necessary installation tasks and locate compatible drivers. Although it is not required to make this determination, Windows uses the function ID tuple internally to determine what type of device is on the PC Card.

12. Configuration table entry tuples listed in priority order

Required

Configuration table entry tuples are placed in the preferred order for configuring the device. Windows processes the tuples in the order they are placed in the CIS. From these tuples, Windows creates a logical configuration in this order and prioritizes them in the same order. Notice that for multiple voltage cards, the voltage policy is to prioritize 3.3-volt configurations (if supported by the system) over 5-volt configurations, regardless of the order of the configuration table entry tuples (CISTPL_CFTABLE_ENTRY).

13. Card specifies maximum configuration options

Required

Many older PC Cards specified fixed configurations in order to address compatibility with existing software. However, this is not the intended use for tuples; the configuration software should be responsible for compatibility. The tuples should be used only to rule out configurations not supported by the hardware.

If you must provide fixed configurations for an operating system other than Windows, you still must provide one or more entries that specify the maximum configurability that the hardware can handle.

Plug and Play Design for CardBus

This section summarizes the Plug and Play requirements for CardBus cards. CardBus was designed as a combination of the PC Card 16 and PCI. The goal is to gain the benefits of PCI in a PC Card format. Consistent with this goal, Windows support for CardBus places specific requirements on CardBus cards.

14. Configuration Space meets Common Silicon Guidelines

Required

The Common Silicon Guidelines are defined in Section 2.6 of the *PC Card Standard Guidelines, Volume 10.*

The standard for CardBus defines a PCI-like Configuration Space that is not fully compliant with the PCI specification. Specifically, under the CardBus standard, card vendors do not have to implement certain critical fields in the Configuration Space (described in the PC Card standard as allocated). In the PC Card standard guidelines for silicon common to both PCI and CardBus products, the implementation of these fields is recommended.

However, to maintain compatibility with existing PCI system software and drivers for PC 98, Windows will support only CardBus cards whose Configuration Space is designed to meet the Common Silicon Guidelines. This is a requirement because CardBus configuration is performed by the PCI software, which knows how to deal with all aspects of PCI topology configuration, including bridging. Without the allocated fields, the cards cannot be fully treated as PCI devices and therefore cannot be supported under Windows.

The required allocated fields are listed in the following table.

Required Allocated Fields

Field	Description and comments
Vendor ID	This read-only field contains a unique ID (in PCI space) for the card manufacturer. The PCI SIG allocates unique IDs.
Device ID Revision ID	These read-only fields are vendor-assigned values that uniquely identify the device (among all vendors of PCI or CardBus products).
Class Code	This read-only field is defined in PCI 2.1. It describes what type of device the card is.
Max_Lat Min_Gnt	These read-only fields specify the desired settings for Latency Timer values according to PCI 2.1. A value of 0 indicates the device has no major requirements for the settings of Latency Timers.
Interrupt Line	This register must be read-write and must not be connected to anything, just as on PCI cards. This register is used to store the current IRQ routing for the device.

15. RESERVED fields comply with PCI 2.1

Required

The CardBus specification also lists two RESERVED fields (offset 2C in the Configuration Space), which have since been defined in PCI 2.1. These fields are also required on CardBus cards for Windows compatibility.

Required RESERVED Fields

Field	Description
Subsystem ID	If different from Device ID
Subsystem Vendor ID	If different from Vendor ID

16. CardBus card implements required and recommended tuples

Required

For CardBus, Windows also requires the same set of card tuples recommended in the PC Card guidelines, as summarized in the following table.

Required CardBus Tuples

Tuple ID	Tuple code	Comments
04h	CISTPL_CONFIG_CB	—
05h	CISTPL_CFTABLE_ENTRY_CB	—
07h	CISTPL_BAR	—
13h	CISTPL_LINKTARGET	Required as first tuple in PC Card standard.
15h	CISTPL_VERS_1	—
20h	CISTPL_MANFID	—
FFh	CISTPL_END	Required as end-of-chain tuple in PC Card standard.
21h	CISTPL_FUNCID	Recommended in PC Card standard; required for PC 98.

PC 98 Requirements for PC Card

This section summarizes additional PC 98 requirements for PC Card.

Power Management for PC Card

This section summarizes the specific PC 98 power management requirements for PC Card. Power management requirements for specific device classes are defined in the related chapters in Part 4 of this guide.

17. Socket controller complies with device class power management reference specification

Required

This applies for both PC Card 16-only controllers and CardBus controllers.

The *PC Card Controller Device Class Power Management Reference Specification, Version 1.0* or higher, provides class-specific definitions of the OnNow device power states (D0–D3) for these devices. The specification also covers device functionality expected in each power state and the possible wake-up event definitions for the class (for example, whether card insertion should wake the system).

18. PC Card 16 cards implement power-related events using ReqAttn bit and #STSCHG mechanism

Required

Any PC Card 16 card that is capable of signaling a wake-up event to the system (as defined in the device class power management reference specification for its class) must implement the ReqAttn bit and its associated enable bit in the Extended Status register , and must signal on the #STSCHG line.

19. CardBus controllers and cards implement power management specifications

Required

PCI-to-CardBus bridges must comply with the requirements defined in *PCI Bus Power Management Interface Specification, Revision 1.0* or higher. CardBus cards must also comply with the requirements defined in *PCI Bus Power Management Interface Specification, Revision 1.0* or higher.

The CardBus card must use the CSTSCHG pin to signal wake-up events. This is because there is no PME# pin on the CardBus interface, and the CardBus must use PME_EN in the card's Configuration Space to enable wake-up events. Specifically, setting the PME_EN bit in the card's Configuration Space must provide the same behavior as setting both the GWAK and WKUP bits in the card's Function Event Mask register.

For more information about PCI power management specifications, see the "PCI" chapter in Part 3 of this guide.

Device Drivers and Installation for PC Card

This section summarizes PC 98 requirements for PC Card device drivers.

20. No user intervention required for correctly installing devices

Required

The user must not be required to perform any device-installation action other than to insert disks that contain drivers and other files.

21. Device is immediately functional without restarting the system

Required

It must not be a requirement for the user to restart the system in order to begin using the device, either after installation is complete or whenever the device is inserted in the system.

22. ZV-compatible PC Card driver uses DirectDraw LVE

Required

ZV-compatible PC Card drivers must use software interfaces based on 32-bit DirectDraw Live Video Extensions (LVE) in order to configure the graphics controller to receive video input using the ZV port. This includes programming the graphics controller to configure the format of the video data, its location on screen, and so on. LVE is part of DirectX 3.0.

ZV card device drivers must handle dynamic graphics state changes, such as resolution changes, color depth changes, and switching to and from full-screen MS-DOS–based applications.

23. PC Card 16 card driver supports sharing of level-mode interrupts

Required

CardBus systems support both PC Card 16 cards and CardBus cards. In this environment, interrupt sharing becomes an issue because CardBus controllers can use PCI interrupts, which are level-sensitive and sharable. To help alleviate interrupt limitations in CardBus systems, Windows operating systems can take advantage of PCI interrupt-sharing capabilities.

In cases where no ISA IRQs are available to a PC Card 16 card in a CardBus controller, the operating system will assign a PCI interrupt to the card. Therefore, it is strongly recommended that PC Card 16 card drivers are updated to support interrupt sharing. However, it is a PC 98 requirement that PC Card 16 card drivers must “hook” the interrupt (whether sharable or not) before its hardware generates any interrupts.

PC Card References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

CardBus host controllers and Windows compatibility white paper
http://www.microsoft.com/hwdev/busbios/

PC Card Controller Device Class Power Management Reference Specification, Version 1.0
http://www.microsoft.com/hwdev/onnow.htm

PC Card diagnostic utility and white papers
Microsoft provides a diagnostic utility that supports multiple voltage cards, multifunction child cards, and the new device ID mechanism, and that can be used to help design or verify CIS for use with PC Card support under Windows. This utility (Dtpl.exe), plus white papers related to PC Card implementations under Windows operating systems, is available on the web site at http://www.microsoft.com/hwdev/busbios/.

PCI Bus Power Management Interface Specification, Revision 1.0
PCI Local Bus Specification, Revision 2.1 (PCI 2.1)
PCI SIG
Phone: (800) 433-5177
http://www.pcisig.com

PCMCIA standards, including *PC Card Standard Guidelines* and *PCI to PCMCIA CardBus Bridge Register Description* (Yenta specification)
PCMCIA
2635 North First Street, Suite 209
San Jose, CA 95134 USA
Phone: (408) 433-2273
Fax: (408) 433-9558
E-mail: office@pcmcia.org
http://www.pc-card.com/

Windows and Windows NT DDKs
MSDN Professional membership

Checklist for PC Card

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

1. All devices comply with the PC Card standards
Required

2. System and ZV-compatible PC Card 16 cards comply with ZV standard definitions
Required

3. Controller supports industry-standard ExCA register set
Required

4. System maintains mapping of IRQ Routing Register bits to system interrupt vectors
Required

5. IRQ connections can be determined by using the 0805 register
Required

6. CardBus controllers support both ISA and PCI interrupts
Required

7. System supports industry-standard definition for CardBus bridges
Required

8. BIOS initializes CardBus controller in 82365-compatible mode and supports backward compatibility
Recommended

9. CardBus controllers do not share writable PCI Configuration Space bits
Required

10. Each PC Card 16 memory window in CardBus controller has it own page register
Required

11. Card supports required I/O card tuples
Required

12. Configuration table entry tuples listed in priority order
Required

13. Card specifies maximum configuration options
Required

14. Configuration space meets Common Silicon Guidelines
Required

15. RESERVED fields comply with PCI 2.1
Required

16. CardBus card implements required and recommended tuples
Required

17. Socket controller complies with device class power management reference specification
Required

18. PC Card 16 cards implement power-related events using ReqAttn bit and #STSCHG mechanism
Required

19. CardBus controllers and cards implement power management specifications
Required

20. No user intervention required for correctly installing devices
Required

21. Device is immediately functional without restarting the system
Required

22. ZV-compatible PC Card driver uses DirectDraw LVE
Required

23. PC Card 16 card driver supports sharing of level-mode interrupts
Required

PART 4

Device Design Guidelines

C H A P T E R 1 3

I/O Ports and Devices

13

This chapter presents PC 98 requirements and recommendations for I/O ports and devices, including serial and parallel ports, wireless capabilities, and input devices and connectors.

In general, system designers are encouraged to consider solutions such as USB rather than traditional connections for external devices. USB support is required for PC 98 systems, and easy connectivity is important in situations where devices might be interchanged on a regular basis. System designers are encouraged to take advantage of USB.

Legacy and proprietary game-pad solutions are not acceptable for PC 98. Game pads, joysticks, and other input devices must be implemented as USB devices.

Contents

System Requirements I/O Ports and Devices

This section summarizes PC 98 requirements for serial and parallel ports.

1. System includes connection for external serial devices

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

Recommended: USB or PC Card.

This capability can also be provided as a 16550A serial port or as equivalent I/O capabilities in the system. If a legacy serial port is implemented in a PC 98 system, it must meet the requirements defined in this chapter. If two legacy serial ports are implemented, additional requirements are defined.

For Net PC systems and Office PC 98 systems, remote management capabilities must be implemented as defined in *Network PC System Design Guidelines*, attached as Appendix E in the References part of this guide.

2. System includes connection for external parallel devices

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

Recommended: USB, IEEE 1394, or PC Card.

This connection can also be provided as a parallel port with ECP-mode capabilities. If a legacy port is implemented in a PC 98 system, it must meet the requirements defined in this chapter.

For an Office PC 98 system or a Net PC system, if a parallel port is present, it must be implemented as an ECP-mode parallel port, and remote management capabilities must be implemented as defined in *Network PC System Design Guidelines*. On a RISC-based system, the keyboard must work as the input device using the Advanced RISC computing (ARC) interfaces.

3. System includes external connection for keyboard

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

Recommended: USB.

Although USB is the preferred solution, this connection can also be implemented as a PS/2-style port or by using wireless capabilities in the system.

Note: For a mobile PC, the required USB port can be used to support the requirement for an external pointing device and keyboard connections. However, two PS/2-style ports can be implemented for the pointing device and keyboard, or a single PS/2-style port can be provided for both the pointing device and the keyboard. If a single PS/2-style port is used, the design must include two separate clocks and two separate data lines, and a special cable must be provided that allows both the external keyboard and pointing device to use the single port.

4. System includes external connection for pointing device

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

Recommended: USB or wireless.

Although USB is the preferred solution, this connection can also be implemented using a PS/2-style port.

See the “System includes external connection for keyboard” requirement earlier in this section for issues related to mobile PCs.

5. System includes USB game pad or joystick

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Recommended; wireless*

This device must support the *USB Human Interface Device Class Specification, Version 1.0* or higher. For more information about requirements for USB peripherals, see the “USB” chapter in Part 3 of this guide.

Important: No devices that use legacy or proprietary ports can be included in a PC 98 system.

6. System includes built-in wireless capabilities

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Recommended*

Wireless capabilities can be provided as built-in capabilities in the system or by using PC Card, IEEE 1394, or USB. If wireless capabilities are included in the system, PC 98 requirements must be met as defined in the “Wireless Component Requirements” section later in this chapter.

7. Devices use USB or external bus connections rather than legacy serial or parallel ports

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Recommended*	*Required*

This will become a requirement for all system types in future versions of these guidelines.

Although legacy LPT and COM ports can be provided on a PC 98 system, no devices except printers should be provided with a system that uses these ports. For PC 98, a legacy serial port cannot be used as the connection for the mouse or modem.

8. All devices meet PC 98 general device requirements

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

These include the requirements for a device ID, automated software-only settings for device configuration, device drivers and Windows-based installation, and icons for external connectors. For more information, see the “Basic PC 98” chapter in Part 2 of this guide.

Serial Port Requirements

Serial ports have been used on computers for decades. In the past, standard baud rates for most serial ports were around 19.2K. Now that systems and peripherals have become more demanding, higher-speed devices are necessary to meet the needs of the newest generation of serial ports.

This section summarizes the hardware design features for PC 98 serial ports. The PC 98 general device requirements are defined in the “System Requirements for I/O Ports and Devices” section earlier in this chapter, including the requirements for a device ID, automated software-only settings for device configuration, device drivers and Windows-based installation, and icons for external connectors.

Non-legacy Serial Port Requirements

This section defines PC 98 requirements for non-legacy implementations in support of serial port capabilities.

9. Serial port meets device class specifications for its bus
Required

As required for all PC 98 devices, a serial port implementation that uses a non-legacy bus must meet the specific device class requirements for that bus.

For example, a USB serial port implementation must comply with all related USB specifications, including:

- *Universal Serial Bus Specification, Version 1.0* or higher (also known as the USB core specification)
- *Universal Serial Bus Device Class Definition for Communication Devices, Version 1.0* or higher

  The "Standard Serial Interface Circuit Emulation" appendix in the *USB Device Class Definition for Communication Devices* specifically addresses serial-port compatibility.

Legacy Serial Port Requirements

This section defines PC 98 requirements for legacy serial ports. Legacy ports are not recommended for PC 98 systems, but if implemented, such ports must meet the requirements defined in this section.

10. Legacy serial port is implemented as 16550A UART or equivalent and supports 115.2K baud
Required

A 16550A buffered Universal Asynchronous Receiver/Transmitter (UART) or equivalent buffered legacy serial port is required to support high-speed communications while reducing the CPU requirements for servicing the device. The device must be able to support 115.2K baud.

11. Legacy serial port supports flexible resource configuration and dynamic disable capabilities

Required

A legacy serial port must provide flexible resource configuration and complete dynamic disable capabilities as defined in the *Plug and Play External COM Device Specification, Version 1.0.*

These are the recommended resource settings for non-PCI devices:

- Four I/O locations for each port, where the standard ISA I/O addresses are 3F8h, 2F8h, 3E8h, 2E8h. Using the standard addresses ensures the proper functioning of software that directly addresses these locations.
- Two IRQ signals, where the standard is programmable interrupt controller-based (PIC-based) IRQ 3, IRQ 4. Using the standard IRQ signals ensures the proper functioning of software written for systems that use standard IRQ signals.

Two IRQs are required for each port. If two serial ports are implemented in the system, the IRQs can be assigned as follows:

- For serial port A: PIC-based IRQ 4 and IRQ 11
- For serial port B: PIC-based IRQ 3 and IRQ 10

An IR adapter port might replace a serial port in a system. In such a case, the IR port should use the resource configuration that would otherwise be assigned to the second serial port.

Notice that, as for all devices, IRQ sharing is required if the minimum resource requirement cannot be met.

12. Conflict resolution for legacy serial port ensures availability of at least one serial port

Required

In the event of an irreconcilable conflict with other serial ports on the system, a legacy serial port must be capable of being disabled by Plug and Play software. This allows at least one of the two conflicting serial ports to operate correctly.

PC 98 Parallel Port Requirements

This section summarizes the basic PC 98 design features for parallel ports and peripherals. Each parallel port on a PC 98 system must meet the requirements listed in this section. The PC 98 general device requirements are defined in the "System Requirements for I/O Ports and Devices" section earlier in this chapter, including the requirements for a device ID, automated software-only settings for device configuration, device drivers and Windows-based installation, and icons for external connectors.

Non-legacy Parallel Port Requirements

This section defines PC 98 requirements for recommended non-legacy implementations to support parallel port capabilities.

13. Parallel port meets device class specifications for its bus
Required

As required for all PC 98 devices, a parallel port implementation that uses a non-legacy bus must meet the specific device class requirements for that bus.

For example, a parallel port implementation that uses USB must comply with all related USB specifications, including the USB core specification and any specific device class specification.

Legacy Parallel Port Requirements

This section defines PC 98 requirements for legacy parallel ports.

14. Flexible resource configuration supported for each parallel port
Required

A legacy parallel port must provide flexible resource configuration following the *Plug and Play Parallel Port Device Specification, Version 1.0b.* Resource requirements must be met for each device of this type on the system. The requirements cannot be split between two ports on the system.

For non-PCI devices, the following are the minimum resource requirements for each parallel port on the system:

- Required: Support ISA I/O addresses of 378h and 278h, plus 3BC or a vendor-assigned I/O address. Using these standard I/O addresses ensures proper functioning of software written for operating systems that directly address these locations.

  Recommended: Map the base I/O address to four additional locations.

- Required: Support PIC-based IRQ 5 and IRQ 7. Using these standard IRQs ensures proper functioning of software written for operating systems that use standard IRQ signals.

  Recommended: Support five additional IRQ signals.
- Required: Support two unique DMA channel selections if the parallel port design supports block data transfers to memory using DMA controllers. Notice also that the DMA function will not work on a parallel port without an IRQ because the end of a DMA transfer is signaled by an interrupt.

To ensure Plug and Play support for resolution of resource conflicts, a full list of options for all possible configuration combinations must be enumerated, including:

- Options for both ECP mode (which requires an I/O address, an IRQ, and a DMA selection) and standard LPT mode (which requires only an I/O address).
- Options that specify only the I/O address, allowing Windows to assign the IRQ and DMA channel.

On Intel Architecture systems, the operating system considers the parallel port base address (/) stored in the first BIOS Data Area (BDA) locations to be LPT1. The address stored in the second location is LPT2, and so on. On RISC-based systems, the information is in the ARC tree. On all ACPI-based systems, the information is obtained through the ACPI tree.

15. EPP support does not use restricted I/O addresses
Required

Some enhanced parallel port (EPP) implementations require eight contiguous I/O ports. If EPP support is implemented, the hardware cannot use the ISA I/O address 3BCh as a base I/O address because VGA devices require use of port 3C0h.

16. Compatibility, nibble mode, and ECP protocols meet IEEE 1284-1994 specifications
Required

Support for a parallel port must include, at minimum, the compatibility-mode and nibble-mode protocols required by the IEEE 1284-1994 specification. This allows other IEEE 1284-compliant devices to be connected without problems.

The port must also support the ECP protocol as defined by IEEE 1284 to allow connections with higher-speed parallel peripherals.

17. Port connectors meet IEEE 1284-I specifications, minimum

Required

IEEE 1284-I–compliant ports use a standard DB25 connector found on existing system parallel port designs. This is called an IEEE 1284-A connector in the specification.

IEEE 1284-II–compliant ports use an IEEE 1284-C connector. This connector is used on both the port and the peripheral device.

The parallel port design must provide enough space between the connectors and the surrounding enclosure to allow for a mating connector, connector shell, and latch assembly. The IEEE 1284 specification recommends an IEEE 1284-C connector for all new ports and devices.

18. IEEE 1284 peripherals have Plug and Play device IDs

Required

The device ID is described fully in the IEEE 1284 specification. All characters in the device identification string must consist only of ASCII values 20h–7Fh. The device identification string consists of a leading zero (0), a hexadecimal value that represents the length of the string, and then a set of fields, in ASCII, with a unique identification string.

For PC 98, in addition to the requirements specified in *Plug and Play Parallel Port Device Specification, Version 1.0b*, the device ID string must contain the following keys, at minimum. The keys are case-sensitive and can be abbreviated in INF files as indicated.

Key	Abbreviated string
MANUFACTURER	MFG
MODEL	MDL
CLASS	CLS
DESCRIPTION	DES

All MANUFACTURER and MODEL key values must remain unique for each manufacturer. All MANUFACTURER, MODEL, CLASS, and DESCRIPTION key values must remain static for a specific unit (that is, ID values do not change for different hardware configurations). For example, a user simply adding a memory module to a printer should not change the MODEL key value reported as part of the device ID. However, if the user adds memory by installing an upgrade kit that requires a different driver or requires the existing driver to behave differently, then changing the MODEL value is acceptable as part of the upgrade installation process.

The CLASS key describes the type of parallel device. The CLASS key can contain the values PRINTER, MODEM, NET, HDC, PCMCIA, MEDIA, FDC, PORTS, SCANNER, or DIGCAM. HDC refers to hard disk controller. MEDIA refers to any multimedia device. FDC refers to floppy disk controller.

The DESCRIPTION key is an ASCII string of up to 128 characters that contains a description of the device the manufacturer wants to have presented if a device driver is not found for the peripheral.

For information about how the system determines the correct peripheral device driver, see the Windows and Windows NT DDKs.

19. Device identification string provides CompatibleID key
Recommended

The CompatibleID (CID) key can provide a value that exactly matches a peripheral name supported by a device driver shipped with Windows. The value must match a value listed in the device's INF file.

Mouse Port and Peripheral Requirements

This section defines the specific PC 98 requirements for pointing-device connections and peripherals. Because the Windows and Windows NT operating systems require a pointing device, a PC 98 system board should include an auxiliary port for an external pointing device (most commonly a mouse). For PC 98, it is recommended that systems designers use the USB port for the connection and also that they consider implementing wireless support for an external pointing device.

The PC 98 general device requirements are defined in the "System Requirements for I/O Ports and Devices" section earlier in this chapter, including the requirements for a device ID, automated software-only settings for device configuration, device drivers and Windows-based installation, and icons for external connectors.

For wireless-capabilities requirements, see the "Wireless Component Requirements" section later in this chapter.

20. Pointing-device connection meets requirements for its bus class
Required

The following requirements must be met, depending on the connection type used in the system. These requirements ensure that all Plug and Play requirements are met and that Microsoft drivers support this device.

If a PS/2-style port is used, the following requirements must be met:

- Comply in full with requirements in *Personal System/2 Specification,* by IBM.
- Use an 8042 chip (or equivalent) to ensure compatibility with Windows. In most cases, the existing 8042 keyboard port is sufficient. The 8042 chip initiates a PIC-based IRQ 12 interrupt when the pointing device is connected to the port.
- Support PCI-based IRQ 12 to ensure the proper functioning of software written for legacy systems that use this IRQ signal.
- Return expected codes, including send ID (0F2h) and response ACK (0FAh), plus 1-byte ID.

If a USB port is used, the following requirements must be met:

- Meet requirements in *USB Specification, Version 1.0* or higher
- Meet requirements in *USB Human Interface Device Class Specifications, Version 1.0* or higher
- Implement minidriver support based on WDM Human Interface Device (HID) class support in the operating system

21. Remote control provides PC 98 minimum support
Recommended
If a remote-control device is provided with a PC 98 system, the range of functions implemented on the device will depend on whether the remote control is designed for the business desktop or for the Entertainment PC 98.

There is no defined list of functions that must be included on a remote-control device, but such a device might provide the following types of functions and buttons:

- Power button that turns devices on and off.
- Start button, such as the Windows logo key, that causes a Start menu to be displayed. For information about the Windows logo key, see the “Keyboard Port and Peripheral Requirements “ section later in this chapter.
- Menu button that causes an application-specific menu to be displayed.
- Help button that causes application-specific Help file to be displayed.
- Directional capabilities, which function similarly to the arrow keys on a keyboard.
- Select button that functions similarly to the ENTER key on a keyboard.

The following functions and buttons can also be considered for a remote-control device used with an Entertainment PC 98 system:

- Television button to select the television as the device that will receive input
- Mute button
- Device control buttons, including Volume Up, Volume Down, Channel Up, Channel Down, Fast Forward, Rewind, Play, Stop, Pause, and Record
- Number keys equivalent to a telephone keypad

Keyboard Port and Peripheral Requirements

The primary input component for a PC is the keyboard. An 8042 microcontroller or its equivalent has traditionally controlled the keyboard connection on the system board. However, USB connections and wireless connections are important design considerations for PC 98 keyboards. Also, these design requirements do not exclude (but do not encourage) implementing a legacy AT-style keyboard port.

This section summarizes the specific PC 98 hardware feature requirements for keyboard ports and peripherals. Some keyboard port requirements differ, depending on the type of port being used.

The PC 98 general device requirements are defined in the “System Requirements for I/O Ports and Devices” section earlier in this chapter, including the requirements for a device ID, automated software-only settings for device configuration, device drivers and Windows-based installation, and icons for external connectors.

For requirements that apply if wireless capabilities are provided for the keyboard, see the “Wireless Component Requirements” section later in this chapter.

22. Keyboard connection meets requirements for its bus class
Required

These requirements depend on the type of connection designed into the system and ensure that all Plug and Play requirements are met and that Microsoft drivers support this device.

If a PS/2-style keyboard port is used, it must meet the following requirements:

- Support IRQ 1 on Intel Architecture to ensure the proper functioning of software written for legacy systems, which expect to use this IRQ signal
- Map the I/O address ports to 60h and 64h
- Return expected scan codes, including send ID (0F2h) and response ACK (0FAh), plus 2-byte ID

If a USB connection is used, it must meet the following requirements:

- *USB Specification, Version 1.0* or higher
- *USB Human Interface Device Class Specifications, Version 1.0* or higher
- Minidriver support based on WDM HID class support in the operating system

If a USB keyboard is the sole keyboard implementation in an Intel Architecture system, it must support the USB Boot Device specification. The system BIOS must provide boot support as specified in the "Basic PC 98" chapter in Part 2 of this guide and as defined in *Universal Serial Bus PC Legacy Compatibility Specification, Version 0.9* or higher.

23. No interference occurs between multiple keyboards
Required

For example, when a mobile PC is connected to a docking station, more than one keyboard can be attached to the system simultaneously. The keyboard ports on a mobile PC and a docking station must be able to resolve conflicts between the two ports when the mobile unit is docked. Windows supports multiple configurations through the registry and will determine which keyboard to enable.

For more information about managing resources and devices for a mobile PC/docking station combination, see the "Mobile PC 98" chapter in Part 2 of this guide.

24. Keyboard includes Windows and Application logo keys

Recommended

The following are requirements for a keyboard design that includes any Windows logo keys:

- The keyboard must be developed according to technical requirements in *New Key Support for Microsoft Windows Operating Systems and Applications*.
- The keyboard must be compatible at the Windows virtual key-code level.
- The keyboard must pass the requirements in the Windows logo key testing software.
- The Windows logo key must function as a modifier (CTRL, SHIFT, or ALT).
- The Windows Flag trademark must be clearly distinguished on the key top according to the guidelines provided in *New Key Support for Microsoft Windows Operating Systems and Applications*.

The following are recommendations for a keyboard design that includes any Windows logo keys:

- Both left and right Windows logo keys are not required in order to offer full functionality under the Windows operating system.
- The Application key can be a dual-function key and can be used to replace the FN key. In this case, a single press-and-release action sends the scan code for the Application key, and holding this key down while pressing another key will modify it to perform the FN function.

Given the crowded nature of compact keyboards on mobile PCs and keyboards that support double-byte characters (such as Japanese-language keyboards), it might be difficult to add three new keys. For mobile PCs, minimal implementation of new keys includes the addition of one Windows logo key and one Application key.

Game Pad Requirements

This section presents the minimum PC 98 requirements for game-control devices.

The PC 98 general device requirements are defined in the "System Requirements for I/O Ports and Devices" section earlier in this chapter, including the requirements for a device ID, automated software-only settings for device configuration, device drivers and Windows-based installation, and icons for external connectors.

25. Device meets USB HID class specification requirements
Required
PC 98 game-control devices and drivers must support the *USB Human Interface Device Class Specification, Version 1.0* or higher.

Wireless Component Requirements

This section summarizes the basic design features for wireless components, provided either as IR or RF adapters. IR solutions are based on communication standards developed by the Infrared Data Association (IrDA). IR solutions are further differentiated as IrDA and Control IR based on the following features:

- IrDA devices perform high-speed, point-to-point, bi-directional, short-range data communications such as file transfers among laptop, Personal Digital Assistant (PDA), and desktop systems.
- Control IR devices simultaneously perform low-speed, longer-range communications between multiple devices and can be implemented as bi-directional devices such as keyboards, multimedia, and game controls or as unidirectional devices such as remote controls. An IrDA-approved Control IR specification is expected in 1997, with first products expected to ship in 1998.

No standard exists for legacy consumer IR, which is in wide use today to control consumer-electronics devices such as televisions, VCRs, CD players, and so on. PC systems designed to control such devices or to be controlled by existing IR remote controls must accommodate the lack of standards by adopting a universal consumer-IR approach. Therefore, no PC 98 protocol recommendations are made for interoperability with legacy consumer-IR devices. Because some IR transceiver manufacturers add support for consumer IR in their IrDA transceivers, PC 98 requirements are specified for using such multiprotocol IR devices in a PC.

Many manufacturers are implementing integrated IR solutions for mobile PCs. Various form-factor and environmental issues have limited the adoption of wireless solutions for desktop PCs, including receiver placement in the office environment and limiting conflicting device signals. USB IR bridging devices or hubs will resolve many of these physical placement issues.

A USB working group is developing guidelines on how USB will interface with both IrDA and Control IR devices, with first products expected to ship around the middle of 1998. These guidelines are expected to address an RF bridge in anticipation of future home RF standards that should emerge and will be required for PC 98 when finalized by the working group.

It is recommended that manufacturers move their wireless designs to incorporate fast IR solutions as soon as possible. Fast IR transmits and receives data at speeds of 1.152 Mb/s and 4.0 Mb/s. Fast IR includes design implementations that improve usability. To minimize interoperability issues, manufacturers need to include Serial IR (SIR) backward compatibility in their fast IR solutions.

Manufacturers who are planning to produce wireless PC peripherals such as keyboards, pointing devices, and joysticks are encouraged to use Control IR instead of RF as the communications medium. The Control IR specification was based on the requirement to support a minimum of four joysticks on systems such as Entertainment PC.

Not using RF will help to minimize RF interference to devices that require between-room communications, such as cordless phones. This recommendation does not apply to wireless PC peripherals used in offices, where RF might be the only viable solution sufficiently diffuse and able to go through barriers, and where it is less important to have an RF connection to a desktop PC in order to support voice and data communications. There might also be a need to use RF in the home environment if IR is not sufficiently diffuse or is unable to bounce around obstacles.

The requirements listed in this section must be met if wireless capabilities are provided in the system. Power management requirements for wireless devices are defined in the "Power Management for I/O Ports and Devices" section later in this chapter.

IR Requirements

An IR implementation is recommended for all system types, but especially for Entertainment PCs. If IR capabilities are included, the system must meet the basic design features for IR components, devices, and systems specified in this section.

Manufacturers who are implementing designs that include Control IR devices are strongly encouraged to join IrDA and to obtain the latest copies of the Control IR specification plus information on the availability of parts and driver software.

26. IR device uses NDIS 5.0 miniport driver

Required

This requirement applies for IrDA devices. An NDIS 5.0, IrDA miniport driver is required for all IR devices. Full documentation and sample source code for building such a miniport driver can be found in the Windows NT 5.0 DDK.

27. IR device meets IrDA specifications

Required

Recommended: Support specifications for both IrDA and Control IR devices.

An IR device must be designed to comply with approved IrDA specifications, including IR data devices, bi-directional IR peripherals (Control IR), and unidirectional IR remotes (Control IR).

If the system is intended to run data transfer applications with other IrDA devices, support of the IrDA specification is required. The emergence of digital still-image cameras with IrDA capability increases the importance of IrDA support in home systems. If the system is intended for the home market, support for Control IR and IrDA is recommended to ensure that the consumer will have the expected IR device interoperability.

28. IR device meets IrDA Control IR specification

Required

The IrDA-approved Control IR specification is expected in 1997, with the first products expected to ship in 1998. If a Control IR implementation is used in a PC 98 system, it must be in compliance with this specification.

29. IR device meets PC 98 bus and port specifications

Required

The requirements for all bus classes are defined in Part 3 of this guide. The Windows operating system includes built-in support for devices that use a serial I/O interface; in this case, a wireless device must also comply with the requirements specified in the “Serial Port Requirements” section earlier in this chapter. A device that uses a parallel port must comply with the requirements specified in the “Parallel Port Requirements” section earlier in this chapter. A USB device must comply with the requirements specified in the “USB” chapter in Part 3 of this guide.

30. IR device meets USB guidelines for interfacing with IrDA and Control IR devices

Required

A USB working group is developing guidelines for how USB is to interface with both IrDA and Control IR devices. When these guidelines are finalized, they will be PC 98 requirements for USB IR implementations.

31. IR device supports flexible resource configuration and dynamic disable capabilities

Required

Resource configuration support is specified for IR adapters. The adapter must provide flexible resource configuration and complete dynamic disable capabilities following the specifications for the bus or legacy port used. Resource configuration requirements are defined in the Plug and Play specification for the bus that the device uses for its connection.

32. System supports standard input speeds of 4 Mb/s

Recommended

Device support for input speeds of 4 Mb/s is strongly recommended for all IrDA devices.

33. System differentiates command streams if transceiver includes legacy consumer IR support

Required

This requirement ensures correct implementation for a system that includes IR support for IrDA and consumer IR devices that use different device signals. A system that uses only a specific IR device protocol will restrict the ability to use multiple input devices and might also restrict other capabilities.

The system needs to address IR coexistence issues if a single IR controller is capable of supporting more than one physical-layer IR protocol transceiver, such as legacy consumer IR and IrDA. Such controllers might support only one protocol at a time if they are using a single communication port connection to the PC’s I/O subsystem. A method for either switching automatically to alternate modes or notifying the IR driver of the need to switch modes is required. If the second alternative is used, then the device manufacturer must publish the hardware interface that the software can use for mode switching.

IrDA and Control IR have been designed to coexist and to share the IR media. Controllers are expected to provide separate data connections into the PC using USB. The IrDA and USB industry associations define guidelines for how to build and interface such devices. For more information, contact the organizations listed in the “References for I/O Ports and Devices” section at the end of this chapter.

RF Recommendations

Support for RF capabilities is optional. If RF is included, the implementation must meet the PC 98 general device requirements. More specific recommendations are expected as RF standards emerge, especially for home PCs.

Manufacturers who are implementing designs that include RF devices are encouraged to join the USB Implementers Forum and other relevant RF industry associations to work with other manufacturers on standardization of RF protocols and media within the computer industry.

The following recommendations are offered to help designers make appropriate choices if RF solutions are being considered in a particular system design:

- **Select a low-power RF alternative.** For relatively short-range wireless devices that cannot use IR, it is possible to use low-power RF. Use an RF solution appropriate to the application. For example, cordless keyboard and track-ball devices that need RF instead of Control IR require a maximum range of only 15 to 20 feet.
- **Provide a method to defeat noise and conflict with other RF devices in the environment.** RF devices should be able to defeat noise such as electromagnetic interference (EMI). Also, programmable channel selection, carrier sensing, or the relatively expensive spread-spectrum or frequency-hopping techniques can be used to share the RF medium with other RF devices that might be in the environment.

  Many of the issues discussed in this recommendation are addressed by the governing regulatory agencies.
- **Obtain separate local certification for the system and for the RF device.** Rules for certifying low-power, short-range, unlicensed RF devices vary greatly from country to country. By configuring the RF device as a system add-on, local certification of the system will not be blocked while waiting for certification of the RF device, which might take longer. Configuring the RF device as a system add-on also enables adding RF support to legacy hardware.

PC 98 Design Features for Ports

This section summarizes PC 98 requirements related to the design initiatives defined in Part 1 of this guide.

Plug and Play and Bus Design for I/O Ports and Devices

The items in this section are PC 98 requirements for Plug and Play capabilities.

34. Each device has a unique Plug and Play device ID

Required

For a non-bus–specific system-board device, there must be a device-specific ID.

Each bus-specific device must have a Plug and Play device ID as required for the bus it uses, as defined in Part 3 of this guide. For example, a PCI device must comply with PCI 2.1 and also must provide a Subsystem ID and Subsystem Vendor ID, as defined in the “PCI” chapter in Part 3 of this guide. A USB device must comply with the *Universal Serial Bus Specification, Version 1.0* or higher, and also must provide a unique ID.

35. Automatic resource assignment and dynamic disable capabilities are supported

Required

The system must be capable of automatically assigning, disabling, and relocating the resources used by this device when necessary, using the method required for the related bus class. When the end user changes this device or adds it to the system, setting of resource assignments must not require changing jumpers or switches on either the adapter or the system board.

In the event of an irreconcilable conflict with other devices on the system, the system must be able to disable the device to prevent the system from stalling. If there is a conflict where more than one port or device of the same type is detected on the system, then one of two methods can be used to resolve it:

- Completely disable the built-in port or device.

  For example, if there is a conflict when a second serial port is added to a desktop system, the expansion card overrides the system-board device. Using this method, the system disables the device on the system board and enables the expansion card only. This is the recommended conflict-resolution method for add-on serial, parallel, Musical Instrument Digital Interface (MIDI), and joystick devices.

Or if an expansion card, such as a display adapter with a built-in pointing-device port, is added to a desktop system that has a system-board pointing-device port, then the expansion card overrides the system-board pointing-device port. Using this method, the system disables the pointing-device port on the system board and only accepts pointing-device input from the expansion card.

- Both ports and devices remain active while resolving any conflict by relocating the resources of one or both devices.

  Using this method, either device can be used. For example, in a docking system, the pointing device on a mobile PC and the pointing device on a docking station can be allowed to share pointing responsibilities. Either pointing device can be used, although the software will use only one.

Note: Fixed (static) resource devices can exist to support standard devices, including the keyboard controller (8042). For a system based on Intel Architecture, these fixed resources are located at I/O addresses under 100h. Standard system-board devices should use their ISA-compatible addresses. For a system based on Intel Architecture, this includes devices with I/O port addresses within the reserved range 0h–0ffh. For more information about legacy resources and ISA-compatible addresses, see the “Legacy Support” appendix in the References part of this guide.

Power Management for I/O Ports and Devices

This section summarizes the specific PC 98 power management requirements for I/O ports and devices.

36. Each device complies with its device class power management reference specification

Required

The related device class power management reference specification applies for each specific type of device. For example, for an input device, the *Input Device Class Power Management Reference Specification* is the relevant specification. These specifications also cover device functionality expected for each power state and possible wake-up event definitions for each class. Power states D0 and D3 are required.

37. Device supports wake-up events

Required for wireless input;
optional for other devices

For PC 98, the ability to cause a wake-up event as defined in the device class power management reference specification is required for wireless input devices. It is optional for other devices.

Device Drivers and Installation for I/O Ports and Devices

This section summarizes PC 98 device driver requirements for I/O ports and devices. The items in this section are requirements for all PC 98 systems.

38. Device drivers and installation meet PC 98 requirements
Required

The manufacturer does not need to supply a driver if a PC 98-compliant driver provided with the operating system can be used. If the manufacturer supplies a driver, the requirements for the device driver and installation are defined in the "Basic PC 98" chapter in Part 2 of this guide.

The basic requirements include driver support for unattended installation and Help file support if special driver parameters are used.

For input devices that use USB connections, driver support must be implemented as a minidriver under the WDM HID support provided in the Windows and Windows NT operating systems.

References for I/O Ports and Devices

The following represents some references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

Device class power management reference specifications
http://www.microsoft.com/hwdev/onnow.htm

IBM Personal System/2 Common Interfaces, Part No. S84F-9809
IBM Personal System/2 Mouse Technical Reference, Part No. S68X-2229
International Business Machines Corporation
IBM Customer Publications Support
Or contact an IBM sales representative

IEEE specifications
ASK*IEEE
Phone: (800) 949-4333
Fax: (212) 310-4091
E-mail: askieee@ieee.org
http://www.ieee.org

Global Engineering Documents
Phone: (800) 854-7179 (US)
(613) 237-4250 (Canada)
(303) 792-2181 (Outside North America)
Fax: (303) 397-2740
ftp://ftp.symbios.com/pub/standards/io/

Infrared Data Association Serial Infrared (SIR) Physical Layer Specification
Available only to IrDA members:
Infrared Data Association
PO Box 3883
Walnut Creek, CA 94598 USA
Phone: (510) 943-6546
Fax: (510) 943-5600
E-mail: irda@netcom.com

New Key Support for Microsoft Windows Operating Systems and Applications
Newkeys.zip (self-extracting zip file)
ftp://ftp.microsoft.com/PerOpSys/Win_News

Plug and Play specifications
http://www.microsoft.com/hwdev/specs/

USB specifications
USB HID Usages Table
USB Implementers Forum
Phone: (503) 264-0590
Fax: (503) 693-7975
http://www.usb.org

Universal Serial Bus PC Legacy Compatibility Specification, Version 0.9
http://www.teleport.com/~usb/data/usb_le9.pdf

Windows NT DDK, Windows DDK, and IR Communications for Windows DDK
MSDN Professional membership

Checklist for I/O Ports and Devices

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

Consumer PC 98	Office PC 98	Entertainment PC 98
1. System includes connection for external serial devices		
Required	*Required*	*Required*
2. System includes connection for external parallel devices		
Required	*Required*	*Required*
3. System includes external connection for keyboard		
Required	*Required*	*Required*
4. System includes external connection for pointing device		
Required	*Required*	*Required*
5. System includes USB game pad or joystick		
Recommended	*Recommended*	*Recommended; wireless*
6. System includes built-in wireless capabilities		
Recommended	*Recommended*	*Recommended*
7. Devices use USB or external bus connections rather than legacy serial or parallel ports		
Required	*Recommended*	*Required*
8. All devices meet PC 98 general device requirements		
Required	*Required*	*Required*
9. Serial port meets device class specifications for its bus		
Required		
10. Legacy serial port is implemented as 16550A UART or equivalent and supports 115.2K baud		
Required		
11. Legacy serial port supports flexible resource configuration and dynamic disable capabilities		
Required		
12. Conflict resolution for legacy serial port ensures availability of at least one serial port		
Required		
13. Parallel port meets device class specifications for its bus		
Required		
14. Flexible resource configuration supported for each parallel port		
Required		
15. EPP support does not use restricted I/O addresses		
Required		
16. Compatibility, nibble mode, and ECP protocols meet IEEE 1284-1994 specifications		
Required		
17. Port connectors meet IEEE 1284-I specifications, minimum		
Required		
18. IEEE 1284 peripherals have Plug and Play device IDs		
Required		

19. Device identification string provides CompatibleID key
Recommended

20. Pointing-device connection meets requirements for its bus class
Required

21. Remote control provides PC 98 minimum support
Recommended

22. Keyboard connection meets requirements for its bus class
Required

23. No interference occurs between multiple keyboards
Required

24. Keyboard includes Windows and Application logo keys
Recommended

25. Device meets USB HID class specification requirements
Required

26. IR device uses NDIS 5.0 miniport driver
Required

27. IR device meets IrDA specifications
Required

28. IR device meets IrDA Control IR specification
Required

29. IR device meets PC 98 bus and port specifications
Required

30. IR device meets USB guidelines for interfacing with IrDA and Control IR devices
Required

31. IR device supports flexible resource configuration and dynamic disable capabilities
Required

32. System supports standard input speeds of 4 Mb/s
Recommended

33. System differentiates command streams if transceiver includes legacy consumer IR support
Required

34. Each device has a unique Plug and Play device ID
Required

35. Automatic resource assignment and dynamic disable capabilities are supported
Required

36. Each device complies with its device class power management reference specification
Required

37. Device supports wake-up events
Required for wireless input;

optional for other devices

38. Device drivers and installation meet PC 98 requirements
Required

C H A P T E R 1 4

Graphics Adapters

This chapter presents the PC 98 requirements and recommendations for graphics adapters.

The key design goal is to ensure that graphics hardware behaves consistently across a wide range of applications. Graphics-related issues for PC 98 are based on the need of the system to provide fast, high-quality graphics rendering. For example, new games applications that require 3-D graphics or advanced graphics applications such as computer-aided design (CAD) demand improved support in graphics hardware.

Requirements for MPEG and DVD playback, video input and capture devices, and display monitors are defined in the “Video and Broadcast Components” and “Monitors” chapters in Part 4 of this guide. Requirements for OpenGL support are defined in the “Workstation PC 98” chapter in Part 2 of this guide.

Contents

System Requirements for Graphics Adapters

This section summarizes the PC 98 system requirements for graphics adapters. For exceptions and guidelines for the internal display on mobile PCs, see the “Mobile PC 98” chapter in Part 2 of this guide.

1. Graphics adapter uses PCI, AGP, or another high-speed bus

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*AGP required*

Recommended: Pipelined Accelerated Graphics Port (AGP) attachment with optional sideband addressing and double-clocked data transfer mode.

Entertainment PC 98 systems must use AGP or an integrated graphics subsystem that meets or exceeds AGP performance levels. For AGP requirements, see the “Plug and Play and Bus Design for Graphics Adapters” section later in this chapter.

2. System uses WC with higher-performance processors

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

Write combining (WC) of successive stores to the frame buffer is a requirement for systems that use Pentium Pro and Pentium II or compatible processors.

3. Primary graphics adapter works normally with default VGA mode driver

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

The default video graphics array (VGA) driver is required for installing the operating system, and the primary adapter must support 4-bit planar VGA mode as described in the Windows DDK. Secondary adapters in multiple monitor configurations do not need to provide VGA compatibility.

4. Adapter and driver support multiple adapters and multiple monitors

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

System expansion buses that allow graphics adapters such as PCI and AGP can support the simultaneous use of more than one graphics adapter in the system. Each graphics adapter can support one or more attached monitors, but this is not a requirement.

The device drivers for each graphics adapter must provide the required support to allow the presence of multiple adapters and multiple monitors. The hardware and BIOS support consist of Plug and Play-related configuration and resource requirements that ensure automatic support for use of more than one graphics adapter and for simultaneous display on two or more monitors. For details, see the "Multiple-Adapter and Multiple-Monitor Support" section later in this chapter.

5. Adapter supports television output if system does not include large-screen monitor

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Required*

Recommended: Support both NTSC and PAL output.

Support for television output is a requirement for Entertainment PC 98 systems that do not include a large-screen entertainment monitor. The ability to connect to and use a standard NTSC or PAL television as a large display surface is key to the ability to deliver realistic television, movie, and game experiences. For mobile PCs, television-output capabilities can be used to enable on-screen presentation graphics in the conference room.

The NTSC system must support 640 × 480 at 60 Hz. The PAL system must support 640 × 480 and 800 × 600 at 50 Hz. For information about the related requirements, see the "PC 98 Television Output Requirements" section later in this chapter. This recommendation might become a requirement in the future for Consumer PC systems.

If there is a large-screen monitor included with the system, television output support is not required. For information about requirements for large-screen monitors, see "Monitors" in Part 4 of this guide.

Graphics Adapters Basic Features

This section defines basic PC 98 feature requirements for graphics adapters.

6. Adapter meets PC 98 general device requirements

Required

This includes the PC 98 requirements for a Plug and Play device ID, automated software-only settings for device configuration, device drivers and Windows-based installation, and icons for external connectors. For more information, see the "Basic PC 98" chapter in Part 2 of this guide.

7. Screen resolution and local memory capacity meet PC 98 minimum requirements

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

Recommended: 1024 × 768 × 24 bpp and 1280 × 1024 × [8, 15, 16, 24] bpp.

The adapter must support all VESA standard timings for all PC 98 required resolutions, including 640 × 480 × [8, 15, 16, 24] bpp, 800 × 600 × [8, 15, 16, 24] bpp, and 1024 × 768 × [8, 15, 16] bpp.

For a mobile PC's external display support, the graphics subsystem must support the Basic PC 98 requirements. For a mobile PC's internal display, see the requirements in the "Monitors" chapter in Part 4 of this guide.

For any PC 98 system that requires quality rendering of 2-D graphics, the minimum support is double buffering for up to 800 × 600 × 16 bpp. For any PC 98 system that requires 3-D rendering in any fashion, whether using software or hardware acceleration, the minimum requirements are 800 × 600 × 16 bpp double-buffered, Z buffer, and 1.25-MB local texture cache. On AGP systems, there is no requirement for local texture cache.

For future 3-D titles, the high triangle content will make increased demands on graphics bandwidth. High-performance designs for systems such as Entertainment PC 98 or Consumer PC 98 that will support Direct3D applications should provide sufficient 3-D texture access to meet the 3-D performance recommendations defined in the item "Hardware meets PC 98 3-D accelerator performance requirements" later in this chapter.

Texture compression can provide additional effective texture memory, and it also increases the effective memory bandwidth that is available.

Note: When the user selects 1024 × 768 resolution, by default the graphics adapter must use a non-interlaced refresh rate. A graphics adapter can default to 1024 × 768 interlaced mode in either of the following situations:

- The attached monitor is not DDC-compatible and the user has not selected a monitor type in the display control panel.
- The monitor does not support 1024 × 768 non-interlaced mode, as determined from the Extended Display Identification Data (EDID) or registry settings.

8. Adapter meets VESA specifications for ergonomic timing rates

Required

Recommended: 85 Hz for 1024 × 768, non-interlaced.

The graphics adapter must support, at a minimum, the ergonomic timings documented in the current version of *VESA and Industry Standards and Guidelines for Computer Display Monitor Timing* for all resolutions up to 1280 × 1024 that are supported by the monitor. The minimum required support is 75 Hz for 1024 × 768, non-interlaced. Higher scan rates are preferable under standards published by VESA.

9. All supported color depths are enumerated

Required

The driver and INF file must enumerate all modes supported so that applications can choose their preferred color depth. The driver and INF file must follow the following guidelines for enumeration:

- At a minimum, either 5:5:5 or 5:6:5 modes must be supported.
- If only 5:5:5 mode is supported, the driver must also enumerate this as 16-bpp mode. This is required because some applications only look for 16-bpp support and will run in 8-bit mode if they fail to find a 16-bit mode.
- If both 5:5:5 and 5:6:5 modes are supported, both modes are enumerated.

For each color depth supported, color ordering must be implemented as shown in the following list. Color ordering is shown in the following table from the most-significant bit (MSB) to the least-significant bit (LSB.)

Mode	Color ordering
15 bpp	1 undefined, 5 red, 5 green, 5 blue (URRR RRGG GGGB BBBB)
16 bpp	5 red, 6 green, 5 blue (RRR RRGGG GGGB BBBB)
24 bpp	8 red, 8 green, 8 blue (RRRR RRRR GGGG GGGG BBBB BBBB)
32 bpp	8 undefined, 8 red, 8 green, 8 blue (UUUU UUUU RRRR RRRR GGGG GGGG BBBB BBBB)

10. Graphics operations use relocatable registers only

Required

VGA registers must not be used to perform graphics operations such as bit blting, palette setting, and pointer movement. The registers used for these graphics operations can be either I/O locations or memory-mapped locations, but must be relocatable. Normal system operation—except for system startup and mode setting—should never require use of base VGA registers.

DirectDraw and Direct3D functionality must be independent of VGA. This means that graphics require VGA only for initialization.

11. Adapter supports downloadable RAMDAC entries for image color matching

Required

For graphics adapters that support 24-bit or higher displays, downloadable RAM digital-to-analog converter (RAMDAC) entries should be included to perform gamma correction in hardware. This capability supports the related requirement to use VGA only for system initialization.

This capability is also one of the most frequently requested features from independent software vendors (ISVs). For example, it supports transition effects in Internet Explorer 4.0 and other applications.

Image color matching (ICM) uses this capability to ensure that gamma is correct in the monitor and that games applications use this for palette switches.

12. Adapter supports DDC monitor detection

Required

This requirement is based on the *Display Data Channel Standard, Version 2.0* (DDC), which defines the communication channel between the display and host system. The software can use this information to properly manage output to the various displays and to prevent the disabling of television output if no monitor is attached.

Hardware Acceleration for Video Playback

This section presents the PC 98 requirements for graphics adapters support for video playback.

13. Adapter supports video overlay surface with scaling

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

The graphics adapter must be able to support a minimum of one off-screen video overlay surface the has following characteristics:

- **Size.** Support 720 × 480 or larger.
- **Screen Resolutions.** The video overlay must be fully operative at a minimum screen resolution of 800 × 600 at 60 Hz and color depths of 8 bpp and 16 bpp.

  Recommended: Full support at 1280 × 1024 and below, with color depths of 8, 16, 24, and 32 bpp.
- **Color formats.** The required formats include the following:
  - YUV 4:2:2 YUY2: A packed-pixel byte stream for every pixel in the order of Y1, U, Y2, V.
  - YUV 4:2:2 UYVY: A packed-pixel byte stream for every pixel in the order of U, Y1, V, Y2.
  - YUV 4:2:0 YV12: A system-board byte stream for the entire plane in the order of Y plane, V plane, U plane.

  The YUV color space and intensity range are defined by the ITU-R BT.601-4 standard (previously called CCIR-601), where U is CB and V is CR. These formats use less memory while maintaining high quality, and YUV is the native format for many image and video compression standards.
- **Scaling.** Up/down scaling to any size window, with bilinear interpolation (referred to as “hardware arithmetic stretching” in *PC 97 Hardware Design Guide*).

  For Consumer PC 98 and Office PC 98, the down-scaling hardware must support bilinear interpolation or better for size reductions from 1:1 up to a 2:1 ratio. For Entertainment PC 98, bilinear interpolation or better is required for all size reductions. Notice that for down-scaling ratios greater than 2:1, other filtering techniques might provide higher quality images and better performance.

  Recommended: Additional independent and resizable overlays for support of picture-in-picture (PIP) video features and multiple video conferencing windows. For example, the graphics adapter should support overlay using a different color key, or one overlay using a color key while another does not. The support should also allow the overlays to be zoomed independently.

14. Hardware supports VGA destination color keying for video rectangle

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required with DVD-Video*	*Required*

Recommended: Support for alpha-blended overlay per pixel to support DVD-Video subpicture information and enhanced user interface designs.

This is a requirement for video overlays. The compositing of the video plane under the VGA plane with the VGA pixels must be independently controllable for each VGA pixel. This VGA destination color keying must function in all video modes, using either a specific color/color range (on 4-bit, 8-bit, 16-bit, and 24-bit SVGA modes) or additional alpha blending bits in the color plane bits (on 15-bit and 32-bit SVGA modes).

This color keying of the VGA will allow certain VGA pixels to be replaced by the underlying video pixels on a pixel-by-pixel basis. This feature enables VGA video overlays, controls, Windows pop-up menus, dialog boxes, and so on. Color keying must also work at the same time as any vertical/horizontal scaling that is active for the underlying video.

15. Video port meets PC 98 specifications if present on graphics adapter

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

All graphics adapters that use a video port connection or that enable end users to make such a connection to a video device are subject to this requirement.

The video port is a dedicated connection between video devices, such as an MPEG-2 decoder or TV tuner, and the graphics adapter, either implemented as a hard-wired connection on the same board as the graphics adapter or implemented between separate devices using a cable connection.

PC 98 systems are required to use a video port connection for hardware MPEG-2 decoders. It is also recommended that a video port connection be used for television tuners.

For a graphics adapter that includes a video port, for MPEG-2 or otherwise, the following requirements must be met:

- **Accessibility.** The video port must be accessible directly from an on-board decoder. The video port must also be accessible from decoders on separate cards using a cable connection.
- **Autoflipping.** The video port must support automated overlay and video port buffer flip on video port Vsync.
- **IRQ.** The video port must generate an interrupt request (IRQ) when vertical synchronization (Vsync) occurs. The kernel-mode video transport component of DirectDraw 5.0 can use this IRQ to perform autoflips. This capability allows fields to be skipped by the video port and also prevents an irregular synchronization from overwriting its buffers. This also enables capture of video port and vertical blanking interval (VBI) data.
- **Driver.** The driver must support DirectDraw VPE, which provides a key element of video playback support in DirectX 5.0. This support, as documented in the DirectX 5.0 DDK, must be incorporated to ensure that the graphics adapter and video port take advantage of VPE capabilities in the operating system.

  For more information, see the white paper on DirectDraw VPE and kernel-mode video transport at http://www.microsoft.com/hwdev/devdes/.

Recommended: The following hardware design guidelines are recommended to support high-quality video playback:

- **Maximum height.** The graphics adapter should support a register that limits the maximum height of the field that gets written into memory.
- **Separate pitch and start addresses.** The overlay and the video port should support separate pitch and start addresses. This allows the bob algorithm to be used while the video is interleaved, which makes switching between bob and weave modes possible.
- **Standard video port.** It is expected that an advanced video port specification will be standardized by the VESA Video Port committee within the PC 98 time frame. It is strongly recommended that MPEG-2 implementations on all PC systems and retail adapters comply with this standard as soon as possible after standardization. This will become a requirement in future versions of these design guidelines.

For more information about requirements related to video ports, see the “System Requirements for Video and Broadcast Components” section of the “Video and Broadcast Components” chapter in Part 4 of this guide.

16. Adapter supports MPEG-2 motion compensation acceleration

Recommended

For products that use MPEG-2 software decoders, MPEG-2 motion compensation acceleration is recommended. Microsoft plans to support motion compensation acceleration under DirectX in 1998.

Specifically, this recommendation refers, but is not limited, to the following:

- Motion compensation of YUV 4:2:0 planar surfaces (versus YUV 4:2:2 packed pixel surfaces) to decrease system memory bandwidth requirements
- Full-precision motion compensation (for example, use 9 bits for an 8-bit signed error term) to prevent degradation of video quality
- Bus mastering of error terms and vectors to and from AGP memory (versus system memory) to increase memory bandwidth and CPU cache efficiencies

For more guidelines on MPEG-2 video performance and quality, see the “Video and Broadcast Components” chapter in Part 4 of this guide.

Multiple-Adapter and Multiple-Monitor Support

This section defines the PC 98 requirements for ensuring system support for multiple adapters and multiple monitors. This support ensures that if the user adds a second adapter, resources will automatically be available and the operating system can automatically manage multiple display adapters.

The actual implementation a user might employ could be one of the following:

- Multiple adapters added to the PC system
- A single adapter with a single controller supporting two monitors
- A single adapter with multiple controllers supporting multiple monitors
- Any combination of these scenarios

The support planned for both Windows and Windows NT requires multiple-adapter compatibility in both the graphics adapter and its driver. With this new multiple-adapter/multiple-monitor support, a single adapter that supports multiple monitors can display independent screen images. This support is beyond the current simultaneous display features of some mobile PCs, which simultaneously show the same Windows desktop on two monitors.

The operating-system support for multiple adapters and multiple monitors requires allowing any secondary graphics adapters to be enabled in VGA mode, thus requiring that VGA for the previous adapter be temporarily disabled.

The support for multiple monitors also assumes that for monitors attached to the same system, but showing different images simultaneously, the different displays might have differing X,Y resolutions, color depths, refresh rates, and display capabilities.

For technical details about implementing driver support for multiple adapters and multiple monitors, see the Windows NT 5.0 DDK.

17. Extended resources can be dynamically relocated after system boot

Required

To ensure Plug and Play for multiple-adapter/multiple-monitor capabilities, all non-VGA standard display resources (also known as extended resources, such as register sets and so on) must be capable of being dynamically relocated after system boot.

This is an extension of the “Graphics operations use relocatable registers only” requirement earlier in this chapter. It also is an addition to the “General Plug and Play Requirements” section later in this chapter.

18. VGA resources can be disabled by software

Required

A means must be provided to allow a driver to disable its adapter from decoding standard VGA addresses. The purpose of this is to ensure that the adapter is independent of all other graphics adapters in the system. The adapter must remain fully functional without the VGA addresses. See also the “Graphics operations use relocatable registers only” requirement earlier in this chapter.

Hardware Acceleration for 2-D Graphics

This section summarizes PC 98 guidelines related to 2-D DirectDraw graphics features, which can be implemented as hardware acceleration features.

Entertainment PC 98 systems require hardware acceleration for 2-D graphics. Individual items in this section are marked as required or recommended for that system type.

In general, hardware acceleration for 2-D graphics is not required for Office PC 98 systems. The specific items in this section are required or recommended only if a hardware manufacturer chooses to implement hardware acceleration in a graphics adapter. For workstations, it is recommended that the OEM choose graphics adapters that provide hardware acceleration to support high-quality graphics for use with professional graphics applications.

The items for DirectDraw acceleration in this section are presented in order of importance.

19. Frame buffer can be accessed directly by applications

Consumer PC 98	Office PC 98	Entertainment PC 98
Required	*Required*	*Required*

The visible frame buffers must be accessible. It must be possible for applications to perform direct frame buffer accesses at any time, even while asynchronous accelerator operations are being executed. Without this capability, drivers cannot support DirectDraw or Direct3D on Windows NT, and operations on Windows will not be fully robust.

Some hardware keeps the information in its frame buffers in a format that does not correspond to the linear format that is standard in DirectDraw, such as tiling the pixels to exploit the 2-D coherence of image data. If this is the case, the hardware must perform translations so that DirectDraw surfaces that are being accessed directly appear linear. The hardware performing this translation might be a limited resource, but it must be able to perform translations on at least seven DirectDraw surfaces simultaneously. Support for eight surfaces is recommended.

20. Adapter and driver support linear-mapped, low-resolution modes

Consumer PC 98	Office PC 98	Entertainment PC 98
Required	*Required*	*Required*

This is required for all PC 98 system types, including systems with LCD displays. All graphics adapters currently support linear-mapped low-resolution modes. There is minimal additional driver work to support this.

Decreasing the size of the frame buffer decreases the average polygon size and increases the frame rate for a given scene. These additional modes provide support for software rendering games and software Direct3D.

If low-resolution support is implemented in the hardware, the following low-resolution modes are required:

320 × 200 × 16 bpp	320 × 240 × 16 bpp	640 × 400 × 16 bpp
320 × 200 × 8 bpp	320 × 240 × 8 bpp	640 × 400 × 8 bpp

The following low-resolution modes are recommended:

400 × 300 × 16 bpp	512 × 384 × 16 bpp
400 × 300 × 8 bpp	512 × 384 × 8 bpp

Note: In Windows, low-resolution capabilities must not be defined in the registry so that they do not appear in the display control panel. In Windows NT, the control panel automatically filters out these modes.

21. Adapter supports transparent blter

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

This is required for all PC 98 system types. There is no restriction on source size. A transparent blter can perform a blt with a source key transparent color. This assumes that the blter is asynchronous with the host processor.

22. Hardware supports double buffering with no tearing

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

This is required for all PC 98 system types and must be performed in synchronization with the VBI.

The hardware must support a mechanism for swapping buffers without visible artifacts such as "tearing." The mechanism for doing this is at the discretion of the hardware designer, but it should support tear-free double buffering for both full-screen and non-occluded windowed applications.

Notice that the blts must be performed in synchronization with the vertical scan line to avoid tearing. The ability to read the current scan line supports blting or writing to the screen without tearing. Also, in some contexts such as video playback, this support eliminates the need for a back buffer.

For information about the upper limits of resolution to be supported, see the requirement, "Screen resolution and local memory capacity meet PC 98 requirements," earlier in this chapter.

23. Hardware supports programmable blter stride

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Recommended*	*Required*

This is required for all PC 98 system types as part of the required support for textures. A programmable blter stride ensures that Windows can use linear memory. A fixed stride forces Windows to use rectangular memory management, with all the related inefficiencies. It must be possible to specify different strides for the source and destination on blts.

Hardware Acceleration for 3-D Graphics

This section summarizes guidelines related to Microsoft Direct3D technologies that can be implemented as hardware acceleration features. Supporting the items in this section can result in improved performance and improved memory use.

For all PC 98 systems, the graphics subsystem needs to support 3-D acceleration because the new Internet Explorer 4.0 shell for both Windows 98 and Windows NT 5.0 requires support for 3-D graphics. If 3-D acceleration capabilities are not available in the hardware, then it will be performed automatically in the software using Direct3D under both Windows 98 and Windows NT 5.0.

System designs must include 3-D acceleration capabilities in the hardware when the target market for the system uses 3-D graphical applications. For example, all Consumer PC 98 systems should incorporate 3-D hardware acceleration because games and similar applications use demanding 3-D graphics.

Office PC 98 systems designed for use with CAD or graphics-design applications also require 3-D hardware acceleration. However, in these requirements it is assumed that the typical office system does not require hardware acceleration.

24. Hardware supports RGB rasterization

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

In RGB mode under Direct3D, shading across a surface is accomplished by independently interpolating all color components. The following capabilities are required for red-green-blue (RGB) rasterization:

- Support 800 × 600 × 16 bpp at 75 Hz in full-screen, 3-D graphics mode.
- All required features available at the same time; for example, turning off specular highlights in order to enable fog is not acceptable.
- Flat and Gouraud shading.
- MIP-mapped textures.
- Bilinear or better filtered textures (rather than point-sampled), with perspective correction.

- Alpha blending, including support for the following modes defined for Direct3D in the DirectX 5.0 DDK:

Required	Recommended
D3DBLEND_DESTCOLOR	D3DBLEND_BOTHINVSRCALPHA
D3DBLEND_INVDESTCOLOR	D3DBLEND_BOTHSRCALPHA
D3DBLEND_INVSRCALPHA	D3DBLEND_DESTALPHA
D3DBLEND_INVSRCCOLOR	D3DBLEND_INVDESTALPHA
D3DBLEND_ONE	D3DBLEND_SRCALPHASAT
D3DBLEND_SRCALPHA	
D3DBLEND_SRCCOLOR	
D3DBLEND_ZERO	

  For source RGB alpha blending, transparent primitives are blended with the background, but the background transparency is not updated. This method provides good visual accuracy if there are not too many overlapping transparent objects.

- Depth-based fog of an arbitrary color, where depth is defined as distance perpendicular to the screen.
- Support for per-vertex fog.

  Recommended: Support for range-based and table-based fog, where range is the radial distance from the eye point.

- Edge anti-aliasing, which can require a separate call to define the edge.

  Recommended: Polygon anti-aliasing for higher visual quality, which does not require a separate call to define the edge.

- Specular highlighting.

  Recommended: Use of dedicated interpolants.

The Direct3D reference rasterizer in the DirectX 5.0 DDK supports all of these capabilities.

25. Hardware supports multi-texturing

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Required*

Multi-texturing hardware can apply multiple textures to a polygon. The common application of multi-texturing is map-based techniques for diffuse lighting and specular reflections.

To implement this capability, support is required for two or more sets of independent texture coordinates. It is recommended that hardware support the combining of at least two textures in a single pass.

The following texture combination operations are required:

- MODULATERGB: Component-wise multiplication of both texture colors
- MODULATELPHA: Multiply colors of one texture by the alpha of the other
- ADD: Component-wise addition of both textures
- BLEND: Linear combination of textures weighted by a scalar specified in a register or in a polygon alpha

Multi-texturing is used to compute the texture value that participates in the pixel pipeline implemented in Direct3D in DirectX 5.0. It is independent of the alpha blending stage that has existed in Direct3D since its inception.

This technique should work in combination with fog and alpha blending, but need not operate at the same time as other advanced filtering.

For more information, see the paper on multi-texturing and DirectX available on the web site at http://www.microsoft.com/hwdev/devdes/.

26. Hardware supports texture formats

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

For PC 98 hardware that implements 3-D acceleration, the hardware must support palletized textures. Pallet entries use the corresponding nonpalletized formats shown in the following table.

Required	**Recommended**
8-bit palletized	4-bit palletized
1:5:5:5 ARGB	8:8:8:8 ARGB
4:4:4:4 ARGB	0:5:6:5 ARGB
	4:2:2 YUV

27. Hardware complies with texture size limitations

Consumer PC 98	Office PC 98	Entertainment PC 98
Required	*Recommended*	*Required*

MIP mapping requires that textures of size 1 × 1 be supported. To meet PC 98 requirements, a 3-D accelerator must support this lower limit on texture size.

The texture units must support square and non-square power-of-two textures ($2^n \times 2^m$) up to 256 × 256.

Recommended: The texture unit should support non-power-of-two width and height. This enables the texture mapping unit to be used to emulate blts. Also, it is recommended that the texture unit support an upper limit of 2048 × 2048 rather than the required 256 × 256.

28. Hardware supports destination RGB alpha blending

Consumer PC 98	Office PC 98	Entertainment PC 98
Recommended	*Recommended*	*Recommended*

For destination RGB alpha blending, primitives are blended with the background, updating not only the colors in the frame buffer but also a cumulative transparency that can affect the rendering of subsequent primitives.

29. Hardware supports Z comparison modes and Direct3D-compatible formats

Consumer PC 98	Office PC 98	Entertainment PC 98
Recommended	*Recommended*	*Required*

It is required for Entertainment PC 98 systems and recommended for other system types that 3-D hardware support 16-bit minimum, unsigned, lockable Z buffer format and all Z comparison modes.

Hardware that supports Z buffering must support clearing of the Z buffer through the DirectDraw depth-fill blt mechanism. However, DirectX 5.0 enables Z buffers to be cleared at the same time as destination surfaces. It is recommended that PC 98 hardware support simultaneous clearing of color and Z buffers using the DirectX 5.0 mechanism.

30. Hardware meets PC 98 3-D accelerator performance requirements

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Required*

These performance capabilities are required for Entertainment PC 98 systems and recommended for other system types.

The 3-D rendering subsystem should have triangle setup capability implemented in hardware that is capable of processing triangles at a sustained rate in excess of 1 million triangles per second. Each triangle is assumed to be 1 visible pixel in area, front facing, textured, and composed of three vertices, where each vertex contains a diffuse and specular color component. Rendering conditions should be 16 bpp, bilinear textured, Z buffered, and alpha blended. Triangles should be ordered such that the Z check always passes (that is, the current triangle is in front of all previously rendered triangles).

The 3-D rendering subsystem should be capable of filling triangles at a sustained rate in excess of 40 million pixels per second. Each triangle is assumed to be 10,000 visible pixels in area, with the same attributes as described for triangle setup in the previous paragraph. Rendering conditions are also the same as for triangle setup. Supporting 60 million pixels per second is recommended.

PC 98 Television Output Requirements

This section summarizes the key design issues and requirements for television output capabilities, which are recommended for all PC 98 system types and required for any Entertainment PC 98 system that does not include a large-screen entertainment monitor.

The requirements in this section apply only if the television output capability is present on a PC 98 system or on a graphics adapter that supports television output capabilities. Some television output capabilities listed in this section are required for Entertainment PC 98 only.

The required support allows an NTSC or PAL television to be used as a primary or secondary display surface for the Windows operating system and for Windows-based applications. Such a display surface allows more realistic game, video, and multimedia experiences for users who want to use a large-screen television that they already own.

If television output capabilities are provided in a PC 98 system, support is required for either NTSC or PAL standards. NTSC refers to the television standards first developed in the United States and used in Canada, Japan, and Mexico. PAL refers to the television standards first developed in Germany and used in Austria, Belgium Brazil, Denmark, Finland, the Netherlands, Norway, Sweden, Switzerland, and the United Kingdom.

For more information about world television standards, see the web site at http://www.bbc.co.uk/aberdeen/eng_info/.

31. Adapter supports both NTSC and PAL output

Recommended

It is recommended that the television output adapter supports both output standards. If NTSC is supported, then the NTSC system must support 640 × 400 and 640 × 480 at 60 Hz. If PAL is supported, then the PAL system must support 640 × 480 and 800 × 600 at 50 Hz.

Whether either or both output standards are supported, software must be capable of independently enabling and disabling television and VGA output.

Note: For NTSC, the 60-Hz mode described in this section is actually 59.940 Hz.

32. Default boot mode supports appropriate locale

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

PC 98 systems and graphics adapters must enable television output automatically as the primary display if a VGA monitor is not attached, defaulting to modes compatible with television output in the geographic region for which the adapter was localized. NTSC adapters should default to 60-Hz modes; PAL adapters should default to 50-Hz modes. Ideally, an adapter would support both modes and provide a safe means for the default selection to be changed by a user.

33. Adapter supports underscan scaling

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Recommended*	*Required*

For Consumer PC 98 and Entertainment PC 98 systems, the television output adapter must be able to correct horizontal and vertical overscan using hardware scaling. This allows 640 × 480 resolution modes to fit onto NTSC displays and 800 × 600 resolution modes to fit onto PAL displays.

Driver software must be capable of enabling and disabling scaling and also of adjusting scaling for compatibility with a variety of television monitors. As television monitors age, overscan reduces, so less scaling is required.

34. Adapter supports flicker filter

Required

The television output adapter must use multi-line (3-tap minimum) hardware filtering techniques for flicker reduction. Enable, disable, and adjust capabilities for the flicker filter must be software controllable. Also, overscan should be software-enabled when the PC is playing full-screen video.

35. Adapter provides proper termination

Required

Proper termination is required so that optimal picture quality from any connector does not require displays to be attached to other connectors. For example, a VGA monitor must not be required in order for the S-Video output to appear properly.

36. Adapter supports RCA-style composite video and S-Video connectors

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Required*

Compared to RCA-style composite video, S-Video dramatically improves the picture quality of the NTSC or PAL scan converter. This standard is designed to reduce cross talk between chrominance and luminance signals and to increase the luminance bandwidth capability of the television. For a description of this standard, see http://www.bbc.co.uk/aberdeen/eng_info/.

A Solent Club for Amateur Radio and Television (SCART; also called Peritel) connector can be supported for European markets, but is not required for North American markets. Notice that most European television sets have two SCART sockets. One allows stereo audio and composite video I/O switchable to RGB inputs. The other supports switching from composite input to S-Video input. The first can be used to connect a satellite receiver for RGB capability. The other can be used to connect a VCR using S-Video for television output. A camcorder usually has connections for composite video, stereo audio, and S-Video. For a description of the SCART standard, see the web site at http://www.bbc.co.uk/aberdeen/eng_info/.

37. Adapter supports both VGA and television output

Required

Recommended: Simultaneous output to VGA monitor and television.

In addition to television output, the PC 98 system also must support VGA output to ensure that users with existing large-screen VGA monitors can use this output capability.

38. Software supports positioning

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Recommended*	*Required*

Software must be able to program the television output hardware to position the television image in increments of 4 pixels horizontally and 4 scan lines vertically.

39. Software supports detection of television connection

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Recommended*	*Required*

For Consumer PC 98 and Entertainment PC 98 systems, software must be able to read the television output hardware to detect whether a television is attached to S-Video or composite output connectors. (Detection of a VGA monitor is achieved using the separate DDC requirement for graphics adapters and monitors.)

This is required to allow the operating system and graphics drivers to correctly support display output during the startup sequence (for example, determining what resolution and refresh rate to use) and to allow the applications to adjust their user interfaces appropriately to the screen capabilities.

PC 98 Design for Graphics Adapters

This section summarizes requirements related to the PC 98 design initiatives defined in Part 1 of this guide.

Plug and Play and Bus Design for Graphics Adapters

The items in this section summarize PC 98 requirements for Plug and Play and other resource-related and bus-related capabilities. The specifications in this section are required for all PC 98 systems.

Note: See also the “Graphics operations use relocatable registers only” requirement in the “Graphics Adapters Basic Features” section earlier in this chapter.

General Plug and Play Requirements

The requirements in this section ensure easy configuration.

40. Each device has a Plug and Play device ID

Required

The device must have a unique device ID using the format required for its bus. For example, a PCI device must comply with PCI 2.1 and also must provide a Subsystem ID and Subsystem Vendor ID, as defined in the "PCI" chapter in Part 3 of this guide.

Note: Multiple-monitor support allows Display class devices to be initialized independent of the system initialization process. For this reason, system-board and add-on display devices cannot use the VGA BIOS POST routine to populate the Subsystem Vendor ID because the device's POST code might not be executed until later in the process, after device enumeration occurs. For system-board devices, the system BIOS should populate the Subsystem Vendor ID at power on. Add-on display adapters should provide a method for populating the Subsystem Vendor ID at the point when power is applied and the device is initialized to the state that is ready for POST.

41. System supports conflict resolution, VGA compatibility, and extended registers

Required

When the end user changes or adds a graphics adapter to the system, setting resource assignments must not require changing jumpers or switches on either the card or the system board. The system must be able to automatically relocate the resources used by a graphics adapter on the system board when a graphics adapter expansion card is added to the system. In the event of an irreconcilable conflict with other devices on the system, the system must be able to disable one of the adapters in order to prevent the system from stalling.

The system must support the VGA graphics standard for application compatibility and for the Windows clean-boot error-recovery process. If a VGA BIOS exists on the graphics adapter, it must be able to configure its base address to C0000h and one alternate address (minimum) to prevent conflicts.

Extended resources are additional I/O ports, direct-access frame buffers, or data transfer areas on a graphics adapter that use more resources than does standard VGA. The Windows configuration manager must be able to map the resources to avoid conflicts with other system devices. At least one alternate configuration must be provided for each non-VGA display resource in the event of conflict during the IPL boot.

The software drivers and VGA BIOS (if used) must be able to use alternate configuration register addresses. The system must be able to dynamically disable or relocate VGA resources from C0000h. It must also be possible to re-enable these resources upon system reboot or reset.

For additional related requirements for multiple monitor support, see the “Multiple-Adapter and Multiple-Monitor Support” section earlier in this chapter.

BIOS and Option ROM Requirements for Graphics Adapters

The requirements in this section relate to BIOS support for graphics adapters.

42. Chips support linear packed-pixel frame buffer, relocatable above 16 MB

Required

Note: For DirectDraw, the graphics adapter’s chip set must support linear access to the frame buffer by the host.

Windows is optimized for a graphics adapter with a packed-pixel frame buffer at all supported resolutions. Memory-mapped packed-pixel frame buffers also provide a fast and simple interface between Windows and the graphics adapter. The Windows DIB engine provides a very fast display by writing directly to packed-pixel frame buffers, and this architecture requires that the hardware developer write only a small, simple device driver.

For optimized support with Windows, a linear packed-pixel frame buffer is required over a bank-switched frame buffer. Use 32-bit addresses to allow the linear frame buffer to be placed above the 16-MB ISA boundary, which enables a system to be populated with large amounts of RAM.

If memory or other resources conflict with the frame buffer being mapped into a linear address space, the page frame address can be used with minimal degradation of performance.

43. Option ROM supports DDC2B

Required

This requirement does not apply for systems that use RISC-based processors. The option ROM for the graphics adapter must meet current DDC host requirements documented in *Display Data Channel Standard, Version 2.0*, Level B (DDC2B), published by VESA. This standard defines the functions that support the data channel between the graphics adapter and a DDC-compliant monitor.

44. BIOS setup utility provides option to force use of system-board graphics

Recommended

The OEM should provide an option in the system BIOS setup utility to force the system-board graphics device to be used (that is, to ignore and leave disabled any PCI graphics adapters). This option would ensure that a user with a PCI hot-docking system is always able to undock, because the VGA device will be in the mobile unit.

45. BIOS supports large frame buffers for graphics adapters

Required

The BIOS must support large frame-buffer graphics adapters that have up to 256 MB of frame buffers.

AGP Requirements

This section defines the PC 98 requirements for AGP.

46. AGP meets PC 98 implementation guidelines

Required

The following are the PC 98 requirements for AGP cards:

- Compliant with PCI 2.1, including the PC 98 requirements for Subsystem ID and Subsystem Vendor ID.
- Compliant with *Accelerated Graphics Port Interface Specification, Revision 1.0* or higher. This means the card has an AGP capability pointer with a working AGP capability structure that has the following characteristics:
  - A minimum request-queue depth of 1 DWORD (RQ value of 0).
  - A minimum speed of 1*x*.
  - A workable AGP_ENABLE.
  - System implementation of a Graphics Address Remapping Table (GART).
- Compliance with the *PCI Bus Power Management Interface Specification, Revision 1.0* or higher, including the Configuration Space registers and the device state (D*x*) definitions.

Requirements for PCI Graphics Adapters

The requirements in this section apply for graphics adapters that use the PCI bus.

47. PCI graphics device supports IRQ and correctly populates PCI BARs

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Recommended*	*Required*

Proper IRQ support is necessary for optimal support of video playback. The display driver for Windows queries the actual device to find its register locations and so on. The PCI base address registers (BARs) must be populated correctly for this information to be correct in the registry.

On adapters that do not support IRQ, the Interrupt Pin Register (3Dh) should be zero (0).

48. PCI system-board graphics device is not hidden from Plug and Play enumeration

Required

Some current implementations cause problems for supporting multi-monitor capabilities with system-board graphics devices. Some system vendors hide the system-board graphics adapter from the PCI bus when another graphics adapter is detected in the system. The system-board device must disable the PCI device rather than hiding it.

Power Management for Graphics Adapters

This section summarizes the specific power management requirements for graphics adapters.

49. Graphics adapter complies with device class power management reference specification

Required

The *Display Device Class Power Management Reference Specification, Version 1.0* or higher, provides definitions of the OnNow device power states (D0–D3) for display and graphics devices. The specification also covers device functionality expected in each power state and the possible wake-up event definitions for the class, if any. Power states D0 and D3 are required; D1 and D2 are optional for graphics adapters.

50. Graphics adapter complies with VBE/Core 2.0 extensions for power management

Required

The *VESA BIOS Extension Standard/Core Functions 2.0* (VBE/Core 2.0) specification defines extensions to VGA ROM BIOS services for power management.

Device Drivers and Installation for Graphics Adapters

This section summarizes the requirements for graphics adapters. The specifications in this section are required for all PC 98 systems.

For additional related requirements for multiple-monitor support, see the "Multiple-Adapter and Multiple-Monitor Support" section earlier in this chapter.

51. Device drivers and installation meet PC 98 requirements

Required

The manufacturer does not need to supply a driver for a device if the device passes PC 98 compliance testing using a driver provided with the operating system. If the manufacturer supplies a driver, then the requirements for device drivers and installation are defined in the "Basic PC 98" chapter in Part 1 of this guide. The basic requirements include driver support for unattended installation and Help file support if special driver parameters are used.

Note: For Windows, the display driver (.DRV) component that is loaded and called by the Windows Graphics Device Interface (GDI) is a Win16 module.

52. Driver does not bypass any Microsoft-provided system components

Required

The driver must not bypass or patch any Microsoft-provided system components. For Windows, this includes Gdi.exe, Kernel.exe, User.exe, Dibeng.dll, Mmsystem.dll, Ddraw.dll, D3d*.dll, and so on.

For Windows NT, this requirement applies for all files normally installed in the System32 directory, including but not limited to Win32k.sys, Ntoskrnl.exe, Gdi32.dll, User32.dll, and Mcdsrv32.dll.

53. Applications provided with device meet Win32 requirements

Required

Any Windows-based applications provided with the device must meet Microsoft requirements for software compatibility as defined in the Win32 SDK.

54. Driver supports dynamic color bit-depth change

Required

The graphics adapter must operate properly and must not fail when asked by the operating system to change the color depth.

Graphics Adapters References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

Accelerated Graphics Port Interface Specification, Revision 1.0
http://developer.intel.com

Design guidelines for DirectX, multiple-monitor/multiple-adapter support, and white paper on DirectDraw VPE and kernel-mode video transport
http://www.microsoft.com/hwdev/devdes/

Display Data Channel Standard, Version 2.0
VESA BIOS Extension Standard/Core Functions 2.0 (VBE/Core 2.0)
VESA and Industry Standards and Guidelines for Computer Display Monitor Timing
Specification for Video Interface Port, Version 1.1
Video Electronics Standards Association (VESA)
2150 North First Street, Suite 440
San Jose, CA 95131-2029
Telephone: (408) 435-0333
Fax: (408) 435-8225
http://www.vesa.org

Display Device Class Power Management Reference Specification, Version 1.0
http://www.microsoft.com/hwdev/onnow.htm

PCI Local Bus Specification, Revision 2.1 (PCI 2.1)
PCI Bus Power Management Interface Specification, Revision 1.0
Phone: (800) 433-5177
http://www.pcisig.com

Windows, Windows NT, and DirectX DDKs
MSDN Professional membership

World television and S-Video interface standards
http://www.bbc.co.uk/aberdeen/eng_info/

Checklist for Graphics Adapters

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

Consumer PC 98	Office PC 98	Entertainment PC 98
1. Graphics adapter uses PCI, AGP, or another high-speed bus		
Required	*Required*	*AGP required*
2. System uses WC with higher-performance processors		
Required	*Required*	*Required*
3. Primary graphics adapter works normally with default VGA mode driver		
Required	*Required*	*Required*
4. Adapter and driver support multiple adapters and multiple monitors		
Required	*Required*	*Required*
5. Adapter supports television output if system does not include large-screen monitor		
Recommended	*Recommended*	*Required*
6. Adapter meets PC 98 general device requirements		
Required		
7. Screen resolution and local memory capacity meet PC 98 minimum requirements		
Required	*Required*	*Required*
8. Adapter meets VESA specifications for ergonomic timing rates		
Required		
9. All supported color depths are enumerated		
Required		
10. Graphics operations use relocatable registers only		
Required		
11. Adapter supports downloadable RAMDAC entries for image color matching		
Required		
12. Adapter supports DDC monitor detection		
Required		
13. Adapter supports video overlay surface with scaling		
Required	*Required*	*Required*
14. Hardware supports VGA destination color keying for video rectangle		
Required	*Required with DVD-Video*	*Required*
15. Video port meets PC 98 specifications if present on graphics adapter		
Required	*Required*	*Required*
16. Adapter supports MPEG-2 motion compensation acceleration		
Recommended		
17. Extended resources can be dynamically relocated after system boot		
Required		
18. VGA resources can be disabled by software		
Required		

Consumer PC 98	Office PC 98	Entertainment PC 98
19. Frame buffer can be accessed directly by applications		
Required	*Required*	*Required*
20. Adapter and driver support linear-mapped, low-resolution modes		
Required	*Required*	*Required*
21. Adapter supports transparent blter		
Required	*Required*	*Required*
22. Hardware supports double buffering with no tearing		
Required	*Required*	*Required*
23. Hardware supports programmable blter stride		
Required	*Recommended*	*Required*
24. Hardware supports RGB rasterization		
Required	*Required*	*Required*
25. Hardware supports multi-texturing		
Recommended	*Recommended*	*Required*
26. Hardware supports texture formats		
Required	*Required*	*Required*
27. Hardware complies with texture size limitations		
Required	*Recommended*	*Required*
28. Hardware supports destination RGB alpha blending		
Recommended	*Recommended*	*Recommended*
29. Hardware supports Z comparison modes and Direct3D-compatible formats		
Recommended	*Recommended*	*Required*
30. Hardware meets PC 98 3-D accelerator performance requirements		
Recommended	*Recommended*	*Required*
31. Adapter supports both NTSC and PAL output		
Recommended		
32. Default boot mode supports appropriate locale		
Required	*Required*	*Required*
33. Adapter supports underscan scaling		
Required	*Recommended*	*Required*
34. Adapter supports flicker filter		
Required		
35. Adapter provides proper termination		
Required		
36. Adapter supports RCA-style composite video and S-Video connectors		
Recommended	*Recommended*	*Required*
37. Adapter supports both VGA and television output		
Required		
38. Software supports positioning		
Required	*Recommended*	*Required*

Consumer PC 98	Office PC 98	Entertainment PC 98
39. Software supports detection of television connection		
Required	*Recommended*	*Required*
40. Each device has a Plug and Play device ID		
Required		
41. System supports conflict resolution, VGA compatibility, and extended registers		
Required		
42. Chips support linear packed-pixel frame buffer, relocatable above 16 MB		
Required		
43. Option ROM supports DDC2B		
Required		
44. BIOS setup utility provides option to force use of system-board graphics		
Recommended		
45. BIOS supports large frame buffers for graphics adapters		
Required		
46. AGP meets PC 98 implementation guidelines		
Required		
47. PCI graphics device supports IRQ and correctly populates PCI BARs		
Required	*Recommended*	*Required*
48. PCI system-board graphics device is not hidden from Plug and Play enumeration		
Required		
49. Graphics adapter complies with device class power management reference specification		
Required		
50. Graphics adapter complies with VBE/Core 2.0 extensions for power management		
Required		
51. Device drivers and installation meet PC 98 requirements		
Required		
52. Driver does not bypass any Microsoft-provided system components		
Required		
53. Applications provided with device meet Win32 requirements		
Required		
54. Driver supports dynamic color bit-depth change		
Required		

C H A P T E R 1 5

Video and Broadcast Components

This chapter presents the PC 98 requirements and recommendations for video playback, video input and capture devices, and technologies for broadcast-enabled computers.

Specific requirements related to video and broadcast components are defined in the following chapters:

- Requirements related to graphics adapters and television output capabilities are defined in the "Graphics Adapters" chapter in Part 4 of this guide.
- Requirements related to displays are defined in the "Monitors" chapter in Part 4 of this guide.
- Requirements related to digital cameras and other digital image input devices are defined in the "Scanners and Digital Cameras" chapter in Part 4 of this guide.

Contents

Introduction to Video and Broadcast Components

In 1998, video and broadcast television will become integral elements of PC usage, whether for a Consumer, Office, or Entertainment PC. For PC 98, important design issues include:

- Increased quality of video capture and playback, regardless of price point. This includes increased image resolution and increased frame rates.
- Low-latency video delivery, displaying video from both internal and external video devices.
- Easy connectivity and installation for the end user. This requires a reduction in complexity of installation by decreasing the number of components such as power supplies and add-on cards and decreasing the number of user responses required to install a device.
- Implementation of a graphics adapter video port for use by one or more video sources.

DirectShow for Video Support. No functionality will be added to Video for Windows (VfW) in any future version of the Windows and Windows NT operating systems. Support for video playback is provided only under Microsoft DirectShow (formerly ActiveMovie).

New technologies that will make PCs more compelling by integrating them with television are also becoming available. These technologies consist of broadcast components that allow PCs to receive television programming, data services, and new forms of entertainment that blend the two, plus user-interface elements appropriate for use on large-screen display devices such as a progressively scanned display or a television monitor. The new technologies will enable new applications, such as the following:

- By combining the PC, the television, and the Internet, content companies can create new types of programming.
- By using broadcast technology to push multimedia-rich Internet content to consumers, broadcast networks can deliver and store data locally on the PC, reducing the Internet bandwidth bottleneck while improving the consumer’s overall experience.
- By delivering a new set of secure, billable, and scalable data services—such as subscription services for software, electronic news, and entertainment delivery—broadcast services will encourage the creation of new business models.

These technologies, which will be built into the Windows 98 and Windows NT 5.0 operating systems, are based on industry standards such as MPEG-2, Win32, ActiveX, and DirectX. These technologies are also built on current and emerging standards for broadcast networks and Internet protocols, and they enable multicasting as a point-to-many networking standard for network traffic. Broadcast network capabilities provide a transmission infrastructure that can support automatic software and file updates as well as other services.

Consumer Electronics and PCs. The convergence of consumer electronics and personal computing offers new revenue opportunities for participating manufacturers. It also offers the chance for companies from different industries to collaborate on production of new products and services. Companies developing technologies and services that use these components span every industry involved in technology convergence. The related requirements for the elements of broadcast-enabled television are defined in this chapter.

System Requirements for Video and Broadcast Components

This section summarizes the PC 98 requirements for video and broadcast components.

1. System meets PC 98 requirements for DVD-Video and MPEG-2 playback

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required with DVD-Video*	*Required*

Under Windows and Windows NT, operating-system playback support for MPEG-1 is provided through DirectShow. This requirement refers to built-in system support for DVD-Video playback or any other MPEG-2 playback capabilities, whether provided as a hardware decoder, a software decoder, or a combination of the two. This requirement does not apply for Office PC 98 systems that provide DVD-ROM drives for storage purposes only.

Related requirements are defined in the “MPEG-2 Playback Requirements” and “DVD-Video Playback Requirements” sections later in this chapter.

2. System supports PC 98 analog video input and capture capabilities

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Required*

Analog video capture capability is recommended for Consumer PC 98 and Office PC 98 and is required for Entertainment PC 98.

Video input and capture functionality can be implemented as an add-on device or as a direct interface on the system board. If video capture capability is implemented in a PC 98 system, it must meet the requirements defined in the “Video Input and Capture Requirements” section later in this chapter.

For PC 98, all video input sources and capture devices must implement driver support as defined for WDM Stream class in the Windows NT 5.0 DDK.

3. System includes analog television tuner

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Required*

An analog television tuner is required for any Entertainment PC 98 system. This can be implemented as a cable tuner or broadcast tuner. For information about the supporting tuner device, see the “Television Tuner and VBI Capture Requirements” section later in this chapter

4. System includes digital broadcast or satellite subsystem

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Recommended*

If this capability is included in a PC 98 system, the implementation must include a digital broadcast or satellite network adapter, a smart card, and drivers that meet PC 98 requirements as defined in the “Digital Broadcast Television Requirements” section later in this chapter.

5. System includes DTV support

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Required (U.S. only)*

This recommendation represents an effort to realize the full potential of digital television (DTV) based on technical standards defined by the Advanced Television Systems Committee (ATSC) across a range of PCs, hybrid PC/TVs, and DTV appliances. ATSC DTV offers the richness of high-resolution video and high-fidelity audio joined with the interactive content of the PC and the Internet.

It is anticipated that within the 1998–99 time frame, Entertainment PC 98 systems will include support for ATSC DTV. This is consistent with current projections for when DTV broadcasts will begin in major U.S. markets. As with all PC 98 components, compliance testing will begin when all related components are generally available.

Specifications and technical information are available at http://www.atsc.org. Support for ATSC DTV includes meeting hardware and software requirements for a tuner/demodulator, MPEG-2 decode capabilities, and graphics adapters as defined in the “Digital Broadcast Television Requirements” section later in this chapter.

6. Video input, capture, and broadcast device support is based on DirectX foundation class and WDM Stream class

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

The driver for any video or tuner device must use the DirectX foundation class to control all video data. The WDM Stream class must be used to support any data streaming. For information, see the DirectX 5.0 DDK and the Windows NT 5.0 DDK. See also the PC 98 requirements defined in the “Device Drivers and Installation for Video and Broadcast Components” section later in this chapter.

7. Hardware MPEG-2 decoder uses video port for video data

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

For any PC 98 system that includes an MPEG-2 hardware decoder that is not integrated in the graphics accelerator, the decoder must use a video port for piping video data to the graphics adapter’s frame buffer. Systems with MPEG-2 decoders that use the video port of the graphics accelerator must provide a method to disable output. A separate external multiplexor (MUX) meets this requirement.

Systems with multiple video sources that use the video port on the graphics accelerator—including television tuner and capture on the system board or multiple Zoomed Video (ZV) ports—should offer a way to control the flow of video from multiple video sources into a single video port.

For mobile PCs, the ZV standard is available for CardBus peripherals. ZV support must be implemented as defined in *PC Card Standard Guidelines, Volume 10* (PC Card standard). For more information, see the “PC Card” chapter in Part 3 of this guide.

For tri-stated video sources, each source should have a method for arbitrating among multiple devices.

Driver support for any video port implementation must be based on DirectDraw Video Port Extensions (VPE) as defined in the DirectX 5.0 DDK. For more information, see the related “Video port meets PC 98 specifications” requirement in the “Graphics Adapters” chapter in Part 4 of this guide. See also the white paper on DirectDraw VPE and kernel-mode video transport at http://www.microsoft.com/hwdev/devdes/.

8. PCI-based tuners and decoders support bus mastering with scatter/gather DMA

Required

PCI-based hardware must support byte-aligned, multisegment bus master DMA transfers. Devices that are sources (or sinks) for data must be capable of transferring data to or from multiple, non-contiguous host memory buffers that are byte-aligned and odd-sized. The device must support such byte-aligned, odd-sized, non-contiguous buffers using host memory-based buffer transfer descriptors.

This requirement applies for PCI-based MPEG-2 decoders and digital broadcast or satellite television devices. This is required in order to minimize the CPU bandwidth needed to move data from an input source, such as a DVD drive or digital tuner card, to an MPEG decoder.

Because some MPEG-2 packets are 127 bytes long and MPEG-2 streams often contain data bursts and times when data rates are low, bus mastering operations must be able to operate on non-aligned, odd-length data.

Specifically, this means that each stream (with a minimum of eight streams) must have a set of logical buffers (digital broadcast satellite and DVD require a minimum of 16 buffers) composed of physical data segments (with a minimum of 16 + 1 of up to 64K each). Each logical buffer can begin or end on any byte position in physical memory. Thus, the first and last physical data segment can be smaller than a physical memory page (4K), but the intervening segments will be contiguous multiples of the 4K physical-memory page size.

As defined in the "Storage and Related Components" chapter in Part 4 of this guide, DVD drives and other IDE storage controllers and devices must support DMA.

9. Background tasks do not interfere with MPEG-2 playback

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Recommended*	*Required*

This requirement applies to background tasks initiated by applications included with the PC. Video performance should be such that non-foreground tasks—such as downloading a web page or using answering-machine software—should occur without disrupting video playback, including DVD and television. When the user runs an application in the foreground that requires significant system resources, such as a game or video answering machine, the system should degrade gracefully.

For Consumer PC 98, this requirement applies only to applications that are started automatically by the OEM software pre-configuration, such as programs in the Windows/Start Menu/Programs/Startup folder. This guarantees that the video experience “out of the box” is as good or better than consumer television and other A/V components.

For Entertainment PC 98, this requirement applies to all applications included with the system, whether run automatically by the OEM software pre-configuration or run only by the user. This is a critical requirement for Entertainment PC 98 systems, whose users will rely on the PC to perform normal day-to-day operations simultaneously with DVD-Video playback and television.

Specific examples of operations that must not interfere with MPEG-2 playback include the following:

- Answering the telephone to receive voice mail or fax. This applies only to telephony software included with the PC, not third-party software installed by the user. Notice that telephone answering must not be automatically disabled during MPEG-2 playback unless explicitly configured by the user.
- Running scheduled communications tasks such as automatic connection using the modem or ISDN to transfer e-mail and faxes, download cached Internet content, and so on.

Note: Programs that make intensive use of system resources or that are designed for interactive foreground operation are excluded from this requirement. This includes games, video and audio playback, speakerphone, and disk utilities such as error checking, defragmentation, and virus protection.

Notice also that for software decoder implementations, compliance testing for the Entertainment PC 98 requirement will begin in a reasonable time frame after enhanced kernel-mode support is provided for MPEG-2 video playback in the Windows operating system.

10. All components meet PC 98 general device requirements
Required

This includes the basic requirements for a Plug and Play device ID, automated software-only settings for device configuration, device drivers and Windows-based installation, and icons for external connectors. For more information, see the “Basic PC 98” chapter in Part 2 of this guide.

Note: To ensure proper connection by the user between the video graphics array (VGA) monitor, S-Video, and composite cables and connectors, an icon must be added to any external connector using vendor designs or any of the appropriate designs provided in the “Icons” appendix in the References part of this guide.

MPEG-2 Playback Requirements

The requirements in this section apply for MPEG-2 decoders. All requirements apply for both software and hardware decoders or any combination of both unless otherwise noted in a specific requirement. The requirements in this section apply for devices that support playback of an MPEG-2 stream from any source, including DVD, digital or broadcast satellite systems, ATSC DTV, hard drives, and so on.

Any PC 98 system that includes the ability to play back MPEG-2 must meet the requirements listed here to ensure quality playback of MPEG-2 data. A PC with an MPEG-2 playback application and MPEG-2 decoder is an example of a system that must meet these requirements.

For example, an Office PC 98 system that includes a DVD-ROM drive for storage purposes but does not include DVD-Video playback software is exempt from these requirements but must warn users that the feature is not supported.

For decoder driver requirements, see the “Device Drivers and Installation for Video and Broadcast Components” section later in this chapter. For related MPEG-2 audio playback requirements, see the “Audio Components” chapter in Part 4 of this guide.

Important: An MPEG-2 hardware decoder must meet the PC 98 requirements for bus mastering. If a hardware solution is implemented that is not integrated in the graphics accelerator, a video port must also be included as part of the video subsystem, as defined in the “System Requirements for Video and Broadcast Components” section earlier in this chapter.

11. System warns users if it cannot play DVD movies

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required if no DVD-Video*	*Required*

An Office PC 98 system that includes a DVD-ROM drive used only for storage purposes must warn end users that DVD-Video playback is not supported whenever such media is inserted in the drive. All system types that support playback of DVD-Video discs must warn users if a DVD-Video disc from another region is inserted in the drive.

12. MPEG-2 playback meets PC 98 requirements

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required with DVD-Video*	*Required*

All MPEG-2 decoder implementations—whether implemented as hardware, software, or a combination of both—must be capable of the following:

- **MPEG-2 Main Profile at Main Level (MP@ML) playback, with no dropped frames.** Playback requirements include full-frame rate decode of MPEG-2 MP@ML input streams, up to and including the following frame sizes and rates:

720 × 480 at 60 fields per second	720 × 480 at 24 frames per second
720 × 576 at 50 fields per second	720 × 576 at 24 frames per second

  Decoded frame rate is measured at the graphics frame buffer. The actual rate at which video is displayed (or rendered) is covered in the following requirement for smooth frame delivery.
- **Smooth frame delivery.** Video frames must be displayed within one-half frame of the intended display time. This requirement is satisfied for implementations in which video frames are completely rendered into a DirectDraw surface and flipped using the DirectDraw Flip API or a hardware autoflip within one-half frame of the intended display time.

  It is generally accepted in the video broadcast industry that any level of frame slippage or jitter will be noticeable to viewers and will be considered an annoyance compared to the smooth viewing experience provided by standard broadcast television, VCRs, and so on.
- **Rates for decoding and displaying data.** This requires MPEG-2 data rates of up to 9.8 Mb/s for Consumer PC 98 and Office PC 98, and 15 Mb/s for Entertainment PC 98.
- **Synchronized audio and video.** Audio and video must be synchronized to within one video frame.
- **Proper handling of field-based content.** This requirement includes the dynamic field/frame switching capability.
- **No tearing.** This requires proper video buffering, such as double buffering.
- **Correct display of multiple aspect ratio content.** This requirement includes support for Pan Scan and letterbox-formatted content.
- **Closed captioning support.** Line 21 data must be parsed out of the *extension_and_user_data* and made available for decoding for both DVD and digital broadcast satellite line 21 syntax.

- **Output of all remaining frames at the end of the data sequence.** This requirement ensures output of all remaining frames when the decoder receives one of the following:
  - A sequence_end_code message (which differs from an Ipin::EndOfStream() function call)
  - A time discontinuity
- **Splicing MPEG.** Decoders must properly interpret the Closed_Gop flag by dropping B frames before the first I frame after either a data discontinuity is received or the Broken_Link flag is set.

Notice the related video and MPEG-2 support requirements for graphics adapters, such as YUV (4:2:0 and 4:2:2) off-screen overlay surface and up/down bilinear interpolated scaling as defined in the "Hardware Acceleration for Video Playback" section in the "Graphics Adapters" chapter in Part 4 of this guide.

13. Retail adapters with hardware MPEG-2 decoders enable a standard video port connection to the graphics adapter

Required

Retail upgrade adapters sold independent of PC systems can use PCI for video transfer to the graphics adapter but must also enable a video port connection to the graphics adapter.

It is expected that an advanced video port specification will be standardized by the VESA Video Port committee within the PC 98 time frame. It is strongly recommended that MPEG-2 implementations on all PC systems and retail adapters comply with this standard as soon as possible after standardization. This will become a requirement in future versions of these design guidelines.

14. MPEG-2 decoder supports pull-down algorithm

Recommended

An MPEG-2 software or hardware decoder should be able to detect and behave accordingly when 3:2 pulldown (or any other algorithm) is being used to display 24-fps video. The kernel-mode video transport component in DirectDraw 5.0 requires this information from the decoder in order to know when a particular field-skipping algorithm is being used so it knows which fields to skip.

Notice that even if the subsystem supports 3:2 pulldown out of the decoder, the encoder should still be able to receive the 3:2 content.

For more information, see the DirectX 5.0 DDK; see also the article on http://www.microsoft/com/hwdev/devdes/.

DVD-Video Playback Requirements

In addition to the requirements in the previous section, the following requirements apply for systems that provide DVD-Video playback software and hardware. The goal for DVD and other audio/video (A/V) playback is to ensure that the end-user experience is the same or better than with a stand-alone DVD player.

15. DVD decoder driver correctly handles media types, time discontinuity, and decode-rate adjustment
Required

This requirement specifies that the vendor-supplied minidrivers for DVD, MPEG-2, and AC-3 decoders have the following capabilities:

- Use the correct media types, including validation of all format block fields on connection and on every IPin::QueryAccept message.
- Query for IMediaSample2 on every received media sample to test for a time discontinuity bit.
- Adjust the decode rate in response to IPin::NewSegment() calls for video and subpicture.

16. DVD decoder supports subpicture compositing and closed captioning
Required

The system must be capable of displaying subpicture data as well as providing closed-captioning support for all such data stored on the disc. This requires YUV offscreen overlay surface support, as defined in the “Adapter supports MPEG-2 and DVD-Video features” requirement in the “Graphics Adapters” chapter in Part 4 of this guide.

Subpicture streams must be supported as defined in *DVD Specification, Version 1.0,* by Toshiba Corporation.

Note: Alpha blending is required for static menus. However, until alpha blending of YUV surfaces is implemented in graphics adapters, RGB chroma-keying of subpictures and closed-captioning information appearing over motion video is an acceptable alternative, although visually degraded, for 1998.

17. Subpicture decoder correctly handles subpicture properties and other functions

Required

The minidriver for the subpicture decoder must be able to correctly handle the following:

- Must be able to set the subpicture property
- Must be able to turn the subpicture compositing on and off
- Must be able to set the highlight rect parameters

For more information, see the Microsoft DirectX 5.1 SDK and the DirectX 5.0 information in the Windows NT 5.0 DDK.

18. System supports seamless DVD-Video 1.0 navigation

Required

This requirement includes menu navigation, video selection, and language and subpicture track selection in support of the user's ability to navigate DVD-Video discs. Test sources include but are not limited to the following:

- Matsushita Electronics Incorporated (MEI) test disc
- Joe Kane Productions Video Essentials disc

19. System provides a licensed CSS copyright protection scheme

Required

The system must provide a licensed copy scramble system (CSS) implementation and support for CSS encoded DVD-Video discs to ensure proper protection for content produced in accordance with CSS, including regionalization and analog video protection/analog protection system (APS).

To facilitate the authentication process required by this scheme, software is provided as part of the Windows and Windows NT operating system support for DVD. This allows a DVD-ROM drive to authenticate and transfer keys with a CSS decrypter. Windows and Windows NT operating system software will act as the agent to allow either hardware or software decrypters to be authenticated.

For more information about copyright protection requirements, see the "Storage and Related Peripherals" chapter in Part 4 of this guide. For information about CSS or to obtain a CSS license, contact MEI (see http://www.mei.co.jp), or contact the CSS licensing entity when it is established.

Video Input and Capture Requirements

This section summarizes requirements based on new capabilities that support video capture in the Windows 98 and Windows NT 5.0 operating systems. Analog video capture is required for Entertainment PC 98 but not for other PC 98 system types. If this feature is implemented, the requirements in this section must be met.

For requirements related to digital cameras and other digital image input devices, see the “Scanners and Digital Cameras” chapter in Part 4 of this guide.

20. Video input or capture device supports capture of NTSC/PAL picture quality

Required

Video decoders must be capable of decoding 4:3 aspect-ratio, square-pixel 640 × 480/768 × 576 resolution at 30/25 fps at 16 bpp. Decoding of 4:2:2 data format is also required.

21. Analog video capture device outputs video data rate of 3.7 MB per second, minimum

Required

Systems with capture devices must be capable of capturing 3.7 MB per second to disk, providing the user with the ability to capture of 8:1 compression (M-JPEG) of NTSC video.

22. Video input or capture device supports time-code reading

Recommended

Time code is a standard representation of time developed for the video and film industries. Time code is an absolute time format expressed in hours, minutes, seconds, and frames as required by frame-accurate video-editing applications. Separate standards are established for NTSC and PAL systems, as follows:

- For NTSC-based systems, the defining standard is ANSI/SMPTE 12M. Both drop and non-drop frame formats should be supported.
- For PAL-based systems, the defining standard is IEC Publication 461. For film, the defining standard is SMPTE Recommended Practice (RP) 136.

23. Digital video camera uses external bus support

Required

Digital video cameras must provide connectivity using physical wire and driver support to new external buses with isochronous capabilities, particularly USB and IEEE 1394 for high frame-rate devices.

Note: For systems that support video capture and input from multiple sources, it is recommended that the graphics adapter include a video port. For requirements related to implementing a video port, see the “Video port meets PC 98 specifications” requirement in the “Graphics Adapters” chapter in Part 4 of this guide.

Television Tuner and VBI Capture Requirements

This section defines requirements for television tuner capabilities and VBI data capture capabilities in support of the Windows Broadcast Architecture. This architecture is designed to enable a wide rage of data broadcasting services, including the use of decoded data captured from broadcast television signals during the vertical blanking interval (VBI) as well as from video scan lines. An Entertainment PC 98 system must include both television tuner and VBI capture capabilities.

For more information about the operating system Broadcast Architecture and capabilities supported by Windows operating systems, see the Windows NT 5.0 DDK and the white papers available from http://www.microsoft.com/windows/tv/.

Cable and broadcast television data networks can inject digital data into any or all video scan lines, including those usually used for images, in the same way that VBI lines can contain data. In general, a VBI decoder refers to the processing of the raw VBI samples into data packets. VBI capture refers to an analog-to-digital converter (ADC) taking VBI samples of VBI data bits. A VBI decoder takes those samples and performs additional processing to determine bit values from the samples. The VBI decoder also handles specific encoding schemes.

VBI decoding can be performed either in hardware or in software. Software VBI decoders need access to oversampled VBI data. In operation, a VBI decoder is similar to a network adapter, except that the data flow is receive-only.

Notice that some requirements in this section specify support related to NABTS data or other locale-specific formats. Devices designed for locales that support other standards do not have to meet these requirements. However, some requirements specify NABTS as an example data format; in these cases, the device must meet the requirements for relevant locale standards.

Important: A PCI-based tuner and VBI capture or decoder device must meet the PC 98 requirements for bus mastering. A video port is also highly recommended for all implementations, as defined in the “System Requirements for Video and Broadcast Components” section earlier in this chapter.

24. Television tuner supports PC 98 audio and video performance
Required

The audio and video performance capabilities for a television tuner are similar to those defined in the "MPEG-2 Playback Requirements" section earlier in this chapter, including the following:

- Deliver data at full frame rate with smooth delivery and no duplicated frames for both interleaved and progressive video.
- Audio and video playback synchronized to within one video frame.
- Video output quality includes proper handling of field-based content, no tearing, and correct display of multiple aspect ratio content

25. Television tuner includes stereo tuner and supports SAP

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Required*

This requirement includes support for a secondary audio programming (SAP) channel.

For devices designed for use in Europe and South Africa, the device should support Near-Instantaneously Companded Audio Multiplex (NICAM 728) as the standard for digital multichannel sound transmission.

26. VBI capture oversamples VBI data at least four times
Required

To ensure accurate data reception, data transmitted on all lines of the VBI must be oversampled at least four times the NABTS data bit rate (or locale-specific data bit rate). For example, if there are 288 bits of NABTS data on a scan line, approximately 1,152 bits plus the necessary margin (that is, the number required for timing tolerances in the NABTS specification and also for timing uncertainties within the capture hardware) must be captured per scan line.

27. VBI capture detects validity of scan-line data
Required

Each scan line designated for data reception provides a confidence indication about the quality of the received signal (0–100). This is based on the errors encountered in performing the A-to-D. This confidence information will be used to adjust the ranges of decoded scan lines. This QOS will be used to discover the data available on each possible scan line.

28. VBI capture makes VBI data available to the CPU for processing
Required

Raw data samples from VBI lines must go into host memory that can be addressed by the CPU. This data is used to read data encoded into broadcast transmissions, such as closed captioning, NABTS, and Teletext.

To minimize the CPU bandwidth required to process VBI data, the hardware must have the ability to bus master all individual scan lines to different host buffers.

Digital Broadcast Television Requirements

The requirements in this section apply for any type of system that implements a digital broadcast subsystem, whether receiving satellite, cable, or terrestrial broadcasts. Such capabilities are recommended but not required for all PC 98 system types.

Notice that digital broadcast and satellite support under PC 98 includes all the requirements for hardware decoder capabilities and driver support as defined in this chapter, plus support for the DirectX foundation class, as defined in the Windows NT 5.0 DDK.

Important: A digital broadcast receiver must meet the PC 98 requirements for bus mastering. A video port is also highly recommended for all implementations, as defined in the "System Requirements for Video and Broadcast Components" section earlier in this chapter.

29. Digital broadcast card can receive all video, audio, data, and other streams
Required

This can be a receiver for cable or broadcast DTV. The receiver card must be installed on a PCI or IEEE 1394 bus and must provide data tuning, conditional access, de-multiplexing, and other network-specific functions.

The receiver card must be able to receive both normal broadcast network–related information, such as MPEG video, audio, and program guide information, as well as data-stream information.

30. Digital broadcast card can receive full bandwidth from each frequency

Required

The receiver card must be able to receive from and send to the host all information transmitted on any tuner or transponder frequency. For example, if each satellite system transponder has 30 Mb/s of bandwidth, a single-tuner receiver card should be able to transfer all 30 Mb/s of data to the host.

31. Digital broadcast card can receive a minimum of eight simultaneous streams

Required

Recommended: More than eight simultaneous streams.

The receiver card must be able to simultaneously receive and send a minimum of eight streams to the PC on the same carrier frequency. These streams can be of any type, such as eight simultaneous data streams.

These streams, often called service channel IDs (SCIDs) or program IDs (PIDs), are subdivisions of bandwidth on a single tuner frequency.

32. System includes multiple digital broadcast tuner cards

Recommended

The device can also simultaneously receive two or more frequencies. The ability to tune to multiple frequencies results in better concurrent data and video operation. With two tuners, the viewer could watch a video on one frequency and download web pages on the other. This also enables picture-in-picture or multiple data streams on different channels or transponders.

33. Digital broadcast card provides support for legacy conditional access

Required

Cards must support conditional access mechanisms for any subscriptions, pay-per-view events, and other network-specific access-control mechanisms available on the broadcast services for which they are designed.

In many cases, this is a removable smart card that has been paired with code and run on a secure processor on the card.

34. Digital broadcast card provides signal quality and other diagnostic information

Required

The card must be able to self-test and provide diagnostic information such as signal strength, error rate, cable short-circuit events, and the status of any input fuse or circuit breaker. Because these cards are connected to public networks, these capabilities are essential to the carriers who need to diagnose problems in the system.

35. Digital broadcast card supports general-purpose data cryptography

Recommended

The digital broadcast receiver card must be able to provide triple data encryption standard (DES) hardware—or single-DES when restricted by export laws—and RSA public-key secure decryption hardware. Hardware anti-tampering countermeasures must be implemented. This capability is separate from and completely independent of other digital broadcast capabilities.

To allow secure broadcasting of bulk-encrypted data, a secure decryption engine must be implemented in the hardware. This engine must have fast key loading, low data latency, and high data throughput in order to support high-speed data networks.

All DES and RSA private keys must be stored in protected RAM and ROM, respectively, within the device so that it cannot be easily read using physical means. The manufacturer also must sign the RSA public keys, and the RSA signature must be stored within the decryption hardware. Furthermore, this triple-DES decryption hardware must be able to decrypt data at the full rate that a stream can be acquired by broadcast receiver hardware.

36. Digital broadcast card supports substream filtering

Required

The digital broadcast receiver card must be able to filter out unneeded data substreams (sometimes called subSCIDs or subPIDs) in order to reduce bus activity and CPU usage. Substreams allow data broadcasters to dynamically subdivide their broadcast bandwidth among many data streams of differing size.

Substream filtering lets the host specify which substreams it wants to receive and which should be ignored. This avoids unnecessary bus utilization for data streams that will be discarded by the host software.

37. ATSC DTV tuner is fully implemented

Required

If an ATSC DTV tuner is implemented, it must meet the requirements for video and audio compression, packetized data transport structure, and modulation and transmission system as specified in *ATSC Digital Television Standard* (A/53), available at http://atsc.org/.

The tuner should be the same as the analog television tuner for VHF/UHF reception.

38. Stream splitting is supported using DirectShow filters

Recommended

If a hardware solution is implemented, it must be possible to read all data input. Stream splitting can be done on the host CPU using DirectShow filters in the same manner as support is implemented for DVD-Video input data streams.

For current information about the software support planned for this capability, see the related DTV white papers at http://www.microsoft.com/windows/tv/.

39. MPEG-2 decoder and video port support ATSC DTV standard

Required

In a system that supports DTV reception, both the MPEG-2 decoder and the video port must support the final format specifications for layered ATSC DTV in *ATSC Digital Television Standard* (A/53). The estimated minimum is 2xML resolutions and bit rates, which are proposed as 704 × 480 with a 15 Mb/s input bit rate and a 20 million pixels per second output rate. For current format specifications, see the related white papers on the web site at http://www.microsoft.com/windows/tv/.

Formats for this requirement depend on the specific geographical area, and therefore include ATSC 8-Vestigial Side Band (8-VSB), Digital Video Broadcast (DVB), or Orthogonal Frequency Division Multiplexing (OFDM).

This requirement is in addition to the PC 98 requirements for MPEG-2 hardware and playback capabilities defined in the “MPEG-2 Playback Requirements” section earlier in this chapter.

Notice that this requirement is also in addition to the PC 98 requirements for video ports as defined in the “Graphics Adapters” chapter in Part 4 of this guide. As with all PC 98 components, compliance testing will begin when all related components are generally available.

PC 98 Design for Video and Broadcast Components

This section summarizes requirements related to the PC 98 design initiatives defined in Part 1 of this guide.

Plug and Play and Bus Design for Video and Broadcast Components

The items in this section summarize requirements for Plug and Play and other resource-related and bus-related capabilities.

40. Each device has a Plug and Play device ID
Required

Each device must have a Plug and Play device ID as required for the bus it uses, as defined in Part 3 of this guide. For example, a PCI device must comply with PCI 2.1 requirements and must provide a Subsystem ID and Subsystem Vendor ID as defined in the "PCI" chapter in Part 3 of this guide.

For video and broadcast hardware, a device can be implemented as a single function device or as part of a multifunction device. All memory and register resources for this functionality must be distinct and separate from any other functions in the multifunction case.

41. Conflict resolution and dynamic disable capabilities are supported
Required

The operating system must be capable of automatically assigning, disabling, and relocating the resources used by this device when necessary, using the method required for the related bus class. All configuration settings must be capable of being made through software, with no system reboot required.

When the end user changes this device or adds it to the system, setting resource assignments must not require changing jumpers or switches on either the adapter or the system board. In the event of an irreconcilable conflict with other devices on the system, the operating system must be capable of disabling the device to prevent the system from stalling. A disabled device must not claim any resources while disabled.

42. Dependent video device is not independently enumerated
Required

If a video device is implemented as a dependent device on a multifunction adapter, it must not be independently enumerated. Instead, its parent must be responsible for installing and loading its driver and for updating the registry on its behalf.

Device Drivers and Installation for Video and Broadcast Components

This section summarizes the PC 98 requirements for video and broadcast components.

43. Device drivers and installation meet PC 98 requirements

Required

The manufacturer does not need to supply a driver for a device if the device passes PC 98 compliance testing using a driver provided with the operating system. If the manufacturer supplies a driver, then the requirements for the device drivers and installation are defined in the "Basic PC 98" chapter in Part 2 of this guide. The basic requirements include driver support for unattended installation and Help file support if special driver parameters are used.

All video components must use a WDM minidriver instead of a VfW driver. For PC 98, a VfW driver is not compliant with these requirements. For information about WDM driver support, see the Windows NT 5.0 DDK. See also the related articles at http://www.microsoft.com/hwdev/pcfuture/.

Drivers for hardware decoders and for the audio and video subsystems must be implemented as described in the Windows NT 5.0 DDK in order to support DirectShow, DirectDraw 5.0 VPE, and WDM.

44. Software drivers are installed during hardware driver installation

Required

Any additional required device-dependent software such as software codecs or NDIS transports must be installed during the device driver installation routine and must be included in the device INF file.

45. Applications provided with device meet Win32 requirements

Required

Recommended: Video and image editing applications bundled with the device should support DirectShow.

Any Windows-based applications provided with the device must meet software compatibility requirements as defined by the Win32 SDK. Applications installed with the device must use a standard Windows-based installation method as defined in the Win32 SDK.

46. NDIS 5.0 driver provided for digital broadcast receiver

Required

The driver for the digital broadcast receiver must be implemented as an NDIS 5.0 driver. The miniport portion of the driver has the extra interface required for network-specific functions such as tuning, access control, program-guide retrieval, and MPEG data retrieval. Drivers for each card must be supplied by the card vendor or network provider.

For information about NDIS 5.0 driver support, see the Windows NT 5.0 DDK.

Video and Broadcast Component References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

Advanced Television Systems Committee (ATSC) standards
National Association of Broadcasters, (800) 368-5644
Society of Motion Picture and Television Engineers, (914) 761-1100
E-mail: mktg@smpte.org
http://www.atsc.org

ANSI/SMPTE 12M
SMPTE Recommended Practice (RP) 136 and time-code standards
Society of Motion Picture and Television Engineers
595 West Hartsdale Avenue
White Plains, NY 10607-1824
http://www.smpte.org/stds/stsubj.html

DirectDraw VPE and kernel-mode video transport white papers
http://www.microsoft.com/hwdev/devdes/

DTV and broadcast architecture white papers
http://www.microsoft.com/hwdev/pcfuture/bcast1.htm
http://www.microsoft.com/windows/tv/

DVD Specification, Version 1.0, Toshiba Corporation.
http://www.toshiba.com

EIA Standard #ANSI/EIA-516-1988: "Joint EIA/CVCC Recommended Practice for Teletext: North American Basic Teletext Specification (NABTS)."
Electronic Industries Association
2500 Wilson Boulevard
Arlington, VA 22201-3834
http://www.eia.org/

IEC Publication 461
http://www.iec.ch/

Matsushita Electronics Incorporated (MEI) test disc
http://www.mei.co.jp

PC Card Standard Guidelines, Volume 10 (PC Card standards)
PCMCIA
2635 North First Street, Suite 209
San Jose, CA 95134 USA
Phone: (408) 433-2273
Fax: (408) 433-9558
E-mail: office@pcmcia.org
http://www.pc-card.com/

SFF 8090 (Mt. Fuji specification) and other SFF specifications
FaxAccess: (408) 741-1600 (fax-back)
Fax: (408) 867-2115
ftp://fission.dt.wdc.com/pub/standards/SFF/specs/

Video Essentials test disc from Joe Kane Productions, Inc.
http://www.videoessentials.com/

WDM driver support white papers
http://www.microsoft.come/hwdev/pcfuture/

Windows NT DDK, Windows DDK, and DirectX DDK and SDK, including NDIS and broadcast services documentation
MSDN Professional membership

Checklist for Video and Broadcast Components

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

Consumer PC 98	Office PC 98	Entertainment PC 98
1. System meets PC 98 requirements for DVD-Video and MPEG-2 playback		
Required	*Required with DVD-Video*	*Required*
2. System supports PC 98 analog video input and capture capabilities		
Recommended	*Recommended*	*Required*
3. System includes analog television tuner		
Recommended	*Recommended*	*Required*
4. System includes digital broadcast or satellite subsystem		
Recommended	*Recommended*	*Recommended*
5. System includes DTV support		
Recommended	*Recommended*	*Required (U.S. only)*
6. Video input, capture, and broadcast device support is based on DirectX foundation class and WDM Stream class		
Required	*Required*	*Required*
7. Hardware MPEG-2 decoder uses video port for video data		
Required	*Required*	*Required*
8. PCI-based tuners and decoders support bus mastering with scatter/gather DMA		
Required		
9. Background tasks do not interfere with MPEG-2 playback		
Required	*Recommended*	*Required*
10. All components meet PC 98 general device requirements		
Required		
11. System warns users if it cannot play DVD movies		
Required	*Required if no DVD-Video*	*Required*
12. MPEG-2 playback meets PC 98 requirements		
Required	*Required with DVD-Video*	*Required*
13. Retail adapters with hardware MPEG-2 decoders enable a standard video port connection to the graphics adapter		
Required		
14. MPEG-2 decoder supports pull-down algorithm		
Recommended		
15. DVD decoder driver correctly handles media types, time discontinuity, and decode-rate adjustment		
Required		
16. DVD decoder supports subpicture compositing and closed captioning		
Required		
17. Subpicture decoder correctly handles subpicture properties and other functions		
Required		

Consumer PC 98	Office PC 98	Entertainment PC 98
18. System supports seamless DVD-Video 1.0 navigation		
Required		
19. System provides a licensed CSS copyright protection scheme		
Required		
20. Video input or capture device supports capture of NTSC/PAL picture quality		
Required		
21. Analog video capture device outputs video data rate of 3.7 MB per second, minimum		
Required		
22. Video input or capture device supports time-code reading		
Recommended		
23. Digital video camera uses external bus support		
Required		
24. Television tuner supports PC 98 audio and video performance		
Required		
25. Television tuner includes stereo tuner and supports SAP		
Recommended	*Recommended*	*Required*
26. VBI capture oversamples VBI data at least four times		
Required		
27. VBI capture detects validity of scan-line data		
Required		
28. VBI capture makes VBI data available to the CPU for processing		
Required		
29. Digital broadcast card can receive all video, audio, data, and other streams		
Required		
30. Digital broadcast card can receive full bandwidth from each frequency		
Required		
31. Digital broadcast card can receive a minimum of eight simultaneous streams		
Required		
32. System includes multiple digital broadcast tuner cards		
Recommended		
33. Digital broadcast card provides support for legacy conditional access		
Required		
34. Digital broadcast card provides signal quality and other diagnostic information		
Required		
35. Digital broadcast card supports general-purpose data cryptography		
Recommended		
36. Digital broadcast card supports substream filtering		
Required		
37. ATSC DTV tuner is fully implemented		
Required		

Consumer PC 98	Office PC 98	Entertainment PC 98

38. Stream splitting is supported using DirectShow filters
Recommended

39. MPEG-2 decoder and video port support ATSC DTV standard
Required

40. Each device has a Plug and Play device ID
Required

41. Conflict resolution and dynamic disable capabilities are supported
Required

42. Dependent video device is not independently enumerated
Required

43. Device drivers and installation meet PC 98 requirements
Required

44. Software drivers are installed during hardware driver installation
Required

45. Applications provided with device meet Win32 requirements
Required

46. NDIS 5.0 driver provided for digital broadcast receiver
Required

C H A P T E R 1 6

Monitors

This chapter presents the PC 98 requirements and recommendations for display monitors.

Requirements for graphics adapters and television output capabilities are defined in the “Graphics Adapters” chapter in Part 4 of this guide. For issues related to liquid crystal displays (LCDs), see the “Mobile PC 98” chapter in Part 2 of this guide.

Contents

Design Note for Dot-Pitch Limits

Dot-pitch requirements are not specified for PC 98 because dot pitch depends on resolution and size. Also, design features other than dot pitch contribute to usability for PC applications, such as focus and phosphor. However, the following table defines limits based on monitor size.

800 × 600 Dot-Pitch Limits

Monitor size (inches)	Actual size (inches)	Horizontal size (inches)	Vertical size (inches)	Maximum dot pitch (mm)
13.00	11.74	9.39	7.04	0.30
14.00	12.72	10.18	7.63	0.32
15.00	13.70	10.96	8.22	0.35
17.00	15.66	12.53	9.40	0.40
21.00	19.58	15.66	11.75	0.50
25.00	23.50	18.80	14.10	0.60
27.00	25.46	20.37	15.28	0.65
31.00	29.38	23.50	17.63	0.75
33.00	31.34	25.07	18.80	0.80
35.00	33.30	26.64	19.98	0.85
37.00	35.26	28.21	21.16	0.90

Monitor Basic Features

This section summarizes the basic PC 98 design requirements for mobile, desktop, and entertainment monitors.

1. Color monitor is DDC2B-compliant with unique EDID identifier

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

A monitor designed for or included with a PC 98 system must be compliant with *Display Data Channel Standard, Version 2.0,* Level B (DDC2B), which defines the communications channel between the display and host system.

The monitor also must transmit an EDID structure containing unique ID Manufacturer Name and ID Product Code identifiers, plus all required fields as defined in Section 3 of *Extended Display Identification Data Standard, Version 2.0, Revision 1.0* or higher.

2. Monitor supports ICC color matching

Required

Windows and Windows NT support using color profiles that comply with the International Color Consortium (ICC) Profile Format specification. The Image Color Matching (ICM) APIs and functionality for Windows and Windows NT are described in the Win32 SDK and the Windows NT 5.0 DDK.

For PC 98, color-capable devices such as desktop monitors, printers, scanners, still-image cameras, LCDs, color plasma displays, or other flat-panel devices are required to install one or more ICC profiles for ICC color matching. A monitor color-calibration utility is recommended for generating, editing, and installing ICC profiles. The sRGB profile will be distributed in Windows and Windows NT.

This is also a PC 98 requirement for LCDs, color plasma displays, and other flat-panel devices.

3. Monitor meets all PC 98 general device and driver requirements

Required

This includes the basic requirements for a Plug and Play device ID, automated software-only settings for device configuration, device drivers and Windows-based installation, and icons for external connectors. For more information, see the “Basic PC 98” chapter in Part 2 of this guide.

The manufacturer does not need to supply a driver if a PC 98-compliant driver provided with the operating system can be used. If the manufacturer supplies a driver, the requirements for the device driver and installation are defined in the “Basic PC 98” chapter in Part 2 of this guide. The PC 98 requirements include driver support for unattended installation and Help file support if special driver parameters are used.

Note: Monitor support for Windows is installed using a monitor INF file, as defined in the Windows and Windows NT DDKs.

Desktop Monitor Requirements

This section lists the PC 98 hardware requirements and features for desktop monitors.

4. Monitor meets minimum graphics resolution, based on monitor size
Required

With the following higher resolutions, a larger desktop area can be displayed, more applications can be shown on the display at once, individual windows can be larger, applications can be fully displayed side by side, and so on.

- 14-inch to 15-inch external monitor or built-in mobile PC display = 800 × 600
- 17-inch external monitor or 13-inch to 15-inch LCD = 1024 × 768, non-interlaced
- 19-inch to 21-inch monitor = 1280 × 1024, non-interlaced

Note: These specific monitor sizes are not listed as recommended or required; they merely show the expected resolution.

5. Monitor supports ergonomic timing standards
Required

Recommended: 85 Hz for 1024 × 768.

The monitor must, at a minimum, support the ergonomic timings documented in *VESA and Industry Standards and Guidelines for Computer Display Monitor Timing Version 1.0, Revision 0.6* or higher, for all resolutions supported by the monitor (based on monitor size, as cited earlier in this section). The standards ensure a clear, flicker-free display for traditional PC computing.

Entertainment Monitor Requirements

The Entertainment PC system requires a picture tube ideal for both PC graphics and television/movie video. This section defines the requirements for large-screen entertainment monitors.

Although an entertainment monitor is not required for Entertainment PC 98, a large-screen monitor that is sold with an Entertainment PC 98 system must meet the requirements defined in this section.

6. Large-screen monitor is 20 inches or larger if included with an Entertainment PC system
Required

Recommended: 31 inches or 33 inches, measured on the diagonal.

7. Large-screen monitor is 16:9 if included with an Entertainment PC system

Recommended

Recommended: Support 1280 × 720 resolution at 60 Hz.

The 16:9 ratio supports the output capabilities for high-definition television (HDTV) display.

8. Entertainment monitor supports 800 × 600 at 60 Hz refresh rate

Required

Recommended refresh rate: An integral multiple of the video frame rate for any mode in which video is displayed full screen.

DVD movies and typical satellite digital broadcasts provide main-level/main-profile MPEG-2 streams, which is the middle level of the five possible levels of MPEG-2–encoded video data. This translates to 720 × 480 × 30 fps for NTSC and 720 × 576 × 25 fps for PAL. Consequently, 800 × 600 is the optimal resolution.

For PAL, this is an integral multiple of 25 Hz. For NTSC, this is an integral multiple of 30 Hz, with an ideal rate of 120 Hz to support 24 to 30 fps content. Because most broadcast video content (such as NTSC or satellite MPEG-2 video and film) is created or adjusted through temporal rescaling or pulldown expressly for 60-Hz television monitors (at a 3:2 ratio), further pulldown to other refresh rates (such as 72 Hz) will introduce unacceptable motion artifacts, such as non-linear screen motion.

9. Entertainment monitor’s host control is DDC2B-compliant, with digitally controlled geometry

Recommended

The host control of the monitor should be managed using DDC2B-compatible adapters and drivers. This recommendation is based on DDC2B, which defines the communications channel between the display and host system.

Geometry control is necessary for adjustment of PC television images and includes the following controls: skew, pin cushion, size, brightness, contrast, and position. If implemented, geometry control must be provided through a software application rather than through dials on the monitor case. Controls must be revealed through a driver with a remote-controllable user interface.

Plug and Play Design for Monitors

The items in this section summarize PC 98 requirements for Plug and Play.

10. External monitor meets DDC2B and EDID standards
Required

This requirement is based on DDC2B, which defines the communications channel between the display and host system, and on the EDID standard, which defines data formats for configuration information. This requirement includes the identification string and other EDID data that the monitor sends to the system.

Use the established standard or (if necessary) detailed timings to indicate the maximum resolution that the monitor will support. Using either the established or standard timings will result in greater flexibility when using detailed timing descriptor blocks.

The following are monitor descriptor definitions:

- **FD (monitor range).** This information is essential for enabling the operating system to calculate the optimal refresh rate for any selected resolution.
- **FC (monitor name).** Up to three detailed timing blocks can be used to incorporate the company and model name. These descriptors will be concatenated for a single string, and the blocks must be used in the order in which they are to be concatenated.
- **FF (monitor serial number).** If provided, this information will be placed into the registry for easy access by asset-management software.

Power Management for Monitors

This section summarizes the specific power management requirements for monitors.

11. Monitor complies with device class power management reference specification
Required

The *Display Device Class Power Management Reference Specification, Version 1.0* or higher, provides definitions of the OnNow device power states (D0–D3) for graphics adapters and monitors. The specification also covers device functionality expected in each power state and the possible wake-up event definitions for the class, if any. Monitors must support the D0, D2, and D3 power states. The D1 power state is optional for monitors.

Monitors References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

Display Data Channel Standard, Version 2.0, Level B (includes VBE/DDC)
Extended Display Identification Data (EDID) Standard, Version 2.0, Revision 1.0
VESA and Industry Standards and Guidelines for Computer Display Monitor Timing, Version 1.0, Revision 0.6
Video Electronics Standards Association (VESA)
2150 North First Street, Suite 440
San Jose, CA 95131-2029 USA
Phone: (408) 435-0333
Fax: (408) 435-8225
http://www.vesa.org

Display Device Class Power Management Reference Specification, Version 1.0
http://www.microsoft.com/hwdev/onnow.htm

International Color Consortium
ICC Profile Format specification
http://www.color.org

Universal Serial Bus Monitor Control Class Specification, Version 1.0
Phone: (503) 264-0590
Fax: (503) 693-7975
http://www.usb.org

Windows NT DDK, Windows DDK, DirectX DDK, and Win32 SDK
MSDN Professional membership

Checklist for Monitors

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

Consumer PC 98	Office PC 98	Entertainment PC 98
1. Color monitor is DDC2B-compliant with unique EDID identifier		
Required	*Required*	*Required*
2. Monitor supports ICC color matching		
Required		
3. Monitor meets all PC 98 general device and driver requirements		
Required		
4. Monitor meets minimum graphics resolution, based on monitor size		
Required		
5. Monitor supports ergonomic timing standards		
Required		
6. Large-screen monitor is 20 inches or larger, if included with an Entertainment PC system		
Required		
7. Large-screen monitor is 16:9, if included with PC system		
Recommended		
8. Entertainment monitor supports 800 × 600 at 60 Hz refresh rate		
Required		
9. Entertainment monitor's host control is DDC2B-compliant, with digitally controlled geometry		
Recommended		
10. External monitor meets DDC2B and EDID standards		
Required		
11. Monitor complies with device class power management reference specification		
Required		

C H A P T E R 1 7

Audio Components

This chapter presents the PC 98 requirements and recommendations for audio devices.

Contents

Introduction to PC 98 Audio

The PC 98 audio basic requirements are designed to identify the baseline operating system and hardware audio support available for existing and emerging multimedia applications. They are also designed to ensure that a minimum audio capability exists across a majority of platforms.

The advanced recommendations describe additional software and hardware features beyond the minimum requirements. These recommendations support vertical applications and provide scalability above the baseline audio capabilities by offering higher compatibility, performance, concurrency, or quality.

WDM and PC Audio. One key to the successful advancement of audio in the PC is WDM Audio class support. The architecture performs all audio processing in kernel mode, which significantly improves latency. WDM provides a method for non-Microsoft code to run in kernel mode, facilitating the development of host algorithms from multiple vendors.

WDM also provides a more complete architecture than previous generations. Code common to all audio hardware on a given bus is now part of the operating system, making for faster development with more consistent results. WDM promises to streamline development through the integration of the Windows and Windows NT driver models. Now, the same driver will work under both operating systems.

Advanced Features Enabled by PCI. Once an audio device moves beyond the basic playback and recording functions of low bandwidth material, a more advanced bus interface is required. Simple audio sample transport over the ISA bus can consume a significant portion of system bandwidth. Features that require more than two streams of audio will overload the ISA bus. Because features such as 3-D positioning use as many as eight streams of audio, a shift to PCI is necessary.

PCI audio accelerators offer another powerful feature because of the high bus bandwidth available. In the digital-ready architecture, PCI devices can perform audio processing and output either to a built-in codec or back into memory for delivery to an external digital audio device. In the external case, the host can perform final mixing and sample rate conversion (SRC) if necessary, and then can send the audio to an external USB or IEEE 1394 device.

In addition, the PCI architecture with a separate audio codec targets high-quality audio, that is, with a dynamic range of greater than 85 dB.

External Digital Audio. USB and IEEE 1394 provide excellent mechanisms for delivering digital audio to external peripherals for high-quality conversion (greater than 85 dB dynamic range) to and from analog. In the near term, the popularity of USB makes it a natural choice. In the long term, the consumer-electronics industry envisions IEEE 1394 transporting audio and video among many devices in a simple, high-performance manner.

PC Audio Transitions

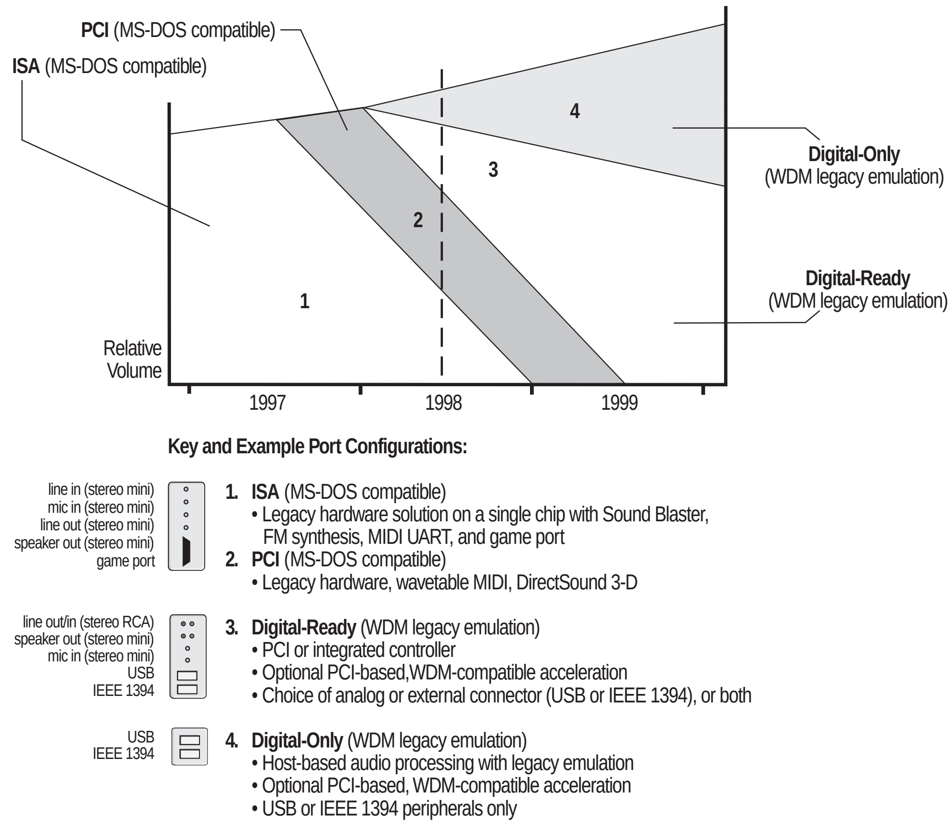
Throughout 1997 and 1998, audio in the PC will offer a wide array of possibilities as a number of transitions take place. Perhaps the most significant transition is the continuing shift from where hardware exclusively processes audio to where the host processor shares some if not all of the burden. To provide the flexibility and performance that is required, a new architecture is needed to connect audio hardware and software. WDM will serve as the support mechanism for the next generation of PC audio.

As the PC is increasingly called upon to play the part of a consumer-electronics device (for example, video-disc playback), sound quality becomes more important. A number of initiatives are underway to achieve optimal sound quality. Another implication of this trend is the need for simpler operation and hardware configuration.

The state of audio functionality is far from stagnant, presenting a challenge for the industry to maximize performance and simplicity, and to add more advanced features. This chapter will focus on how and when this can be done.

The shift to higher quality and support for external digital connectivity will not happen overnight. One objective of PC 98 Audio is to facilitate the transition over the next few years. The remainder of this section is a summary of the projected evolution of PC audio.

The following figure demonstrates that by mid-1998, four viable audio solutions will exist in the PC marketplace. The figure also projects the estimated market share for each solution over the next three years.



Digital Audio Migration

The following explains the marketplace options illustrated in the previous figure:

- **ISA (legacy MS-DOS–compatible).** These are mature, low-cost, single-chip implementations that deliver support for Sound Blaster registers and mixing, Yamaha OPL-compatible frequency modulation (FM) synthesis, MPU 401 MIDI UART, and game port. Each is fully Plug and Play-compatible.
- **PCI (legacy MS-DOS–compatible).** The majority of current PCI implementations are two-chip split digital/analog system-board or add-on designs based on Audio Codec ’97 architecture. Many of these designs incorporate hardware support for legacy ISA devices using PC/PCI DMA, distributed DMA (DDMA), or other similar techniques, and are fully Plug and Play-compatible. Many also support high-quality wave table MIDI synthesis and DirectX acceleration.

- **PCI (digital ready).** The next generation of PCI-based designs have the potential to deliver WDM-compatible audio acceleration that can be part of a filter graph targeting either a built-in audio codec or an external USB codec for output. These designs have the option of leveraging the WDM legacy emulation capabilities or incorporating hardware legacy compatibility. A digital-ready PC can support mixed configuration of analog and/or digital peripherals.
- **USB (digital only).** These systems eliminate all built-in audio rendering resources and use external USB speakers and microphones for audio I/O. Hardware acceleration can be provided by WDM-compatible PCI devices.

The following factors will determine which of the previously listed audio solutions are suitable in 1998:

- Cost. The total bill of materials for ISA, PCI, and USB solutions, including microphone and speakers.
- Compatibility. The desired level of legacy MS-DOS–based game compatibility.
- Quality. The system output dynamic range and total harmonic distortion plus noise (THD+N) performance necessary to meet specific market requirements.
- Features. Support for 48-kHz sample rate operation, which is necessary for DVD.
- Performance. Dependent on the choice of host processor, audio bus (ISA, PCI, or USB), and system audio hardware acceleration features, such as Downloadable Samples (DLS) wave-table MIDI synthesis and multistream Head Related Transfer Function (HRTF) 3-D.
- Connectivity. Internal analog mixer connectivity for legacy CD audio, analog television tuner, line input, or video capture audio input.

Three audio applications merit more detailed discussion and are described in the following sections:

- Scalable audio for 3-D games
- CD and DVD media playback
- Full-duplex H.323/H.324 video and audio conferencing

Scalable Audio for 3-D Games

WDM audio supports the following features specifically for 3-D games under Windows 98 and Windows NT 5.0:

- Software emulation of legacy hardware to support MS-DOS–based games. WDM drivers, which run in kernel mode, provide virtual Sound Blaster Pro, MPU 401, and legacy joystick interfaces.
- A standard interface for the application to provide multiple streams of 3-D–positioned audio. DirectSound 3-D supports software-simulated 3-D (pan and volume) and true HRTF 3-D processing at 22.05 kHz, or hardware acceleration. The architecture supports optimal configuration based on CPU performance and installed hardware.
- A wave-table General MIDI synthesizer entirely in kernel-mode software. This provides 24 voices of music synthesis with 22.05-kHz output. DirectShow, DirectMusic, MMSYS, and virtual MPU 401 can use the synthesizer functions. The architecture supports optimal configuration based on CPU performance and installed hardware.
- A high-quality kernel-mode software SRC capability, which converts data streams (including composite mixes of all 11.025-kHz or 22.05-kHz sources) to the final output mix format, typically 16-bit 44.1 kHz (general SRC support includes other rates).
- A kernel-mode system-wide software mixer, which supports DirectSound, DirectShow, and MMSYS clients, plus kernel-mode WDM filters, including Red Book CD-ROM and MIDI drivers. The mixer implements highly optimized, same sample rate PCM mixing at 8-bit or 16-bit 11.025, 22.05, 44.1, and 48 kHz (general mixing support includes other formats).
- Flexible control of the output destination. The WDM drivers can send the master 16-bit 44.1-kHz or 48-kHz or other format output to an ISA, PCI, USB, or IEEE 1394 audio device. Additionally, support is provided for redirection of PCI-device final-mix output to USB speakers.

The minimum PC 98 audio hardware support necessary for 3-D games is built-in or external audio codec support for playback of 16-bit stereo PCM data at a 44.1-kHz sample rate. For a list of the PC 98 requirements, see the “Basic Audio Requirements” section later in this chapter.

The system designer might choose to include the following optional software or hardware in order to provide additional capabilities:

- Hardware that provides legacy MS-DOS–based interfaces for compatibility with games (Sound Blaster Pro, FM synthesizer, MPU 401, game port)
- Hardware for higher quality or concurrency DLS wave-table MIDI synthesis, with associated mixing and SRC support
- Hardware for higher concurrency HRTF 3-D positional audio, with associated mixing and SRC support
- Digital-ready, WDM-compatible MIDI and HRTF 3-D hardware acceleration

The following table presents actual measurements collected by Intel Architecture Labs (IAL) for scalable audio for 3-D games on Intel Pentium II processors with Intel MMX technology.

Performance Measurements for Scalable Audio for 3-D Games

Function	233-MHz Pentium II processor[1]
Wave-table MIDI synthesis, 24 voices at 22.05 kHz	5%
Mixing with pan and volume for simulated 3-D, eight stereo streams at 22.05 kHz	3%
SRC, upsampling of the stereo composite mix of all 22.05-kHz sources to 44.1 kHz	2%
Audio effects (reverb, tone, or pseudo 3-D), stereo at 44.1 kHz	2%
Audio sample transport to PCI or USB codec	2–4%
Total with simulated 3-D	**14–16%**
HRTF 3-D with cross-talk cancellation for speakers, eight streams at 22.05 kHz	12%[2]
Total with true HRTF 3-D	**23–25%**

[1] Actual performance is dependent on specific processor, chip-set, memory, and I/O subsystem implementations, as well as the software implementation (integer, floating point [FP], or MMX technology). To estimate the performance on other processor speeds with MMX technology, use a ratio based on the processor's MHz rating.

[2] HRTF 3-D performance is based on the 3Q1997 release of Intel 3D Realistic Sound eXperience (RSX); performance of the 1Q1997 3D RSX is 20 percent.

CD and DVD Media Playback

WDM audio supports the following features for CD and DVD media playback under the Windows 98 and Windows NT 5.0 operating systems:

- A kernel-mode CD-ROM driver that emulates MSCDEX commands and implements reading, parsing, and streaming of Red Book CD digital audio to the kernel-mode WDM system-wide mixer at 16-bit 44.1 kHz.
- A Universal Disk Format (UDF) DVD file reader, splitter, and navigator that provides access for DirectShow clients to separate video and audio streams.
- A kernel-mode, system-wide software mixer, which supports DirectSound, DirectShow, and MMSYS clients, plus kernel-mode WDM filters, including Red Book CD-ROM and MIDI drivers. The architecture provides the ability for algorithms from any vendor to decode the DVD audio, and it supports mixing at 16-bit 48 kHz.
- Flexible control of the output destination. The WDM drivers can send the master 16-bit 44.1-kHz or 48-kHz or other format output to an ISA, PCI, USB, or IEEE 1394 audio device. Additionally, support is provided for redirection of the PCI-device final-mix output to USB speakers.

Baseline PC 98 audio hardware support for CD-ROM and DVD media playback requires that the built-in or external audio codec support playback of 16-bit stereo PCM data at either a 44.1-kHz or 48-kHz sample rate. For a list of the PC 98 requirements, see the “Basic Audio Requirements” section later in this chapter.

For MPEG content, the system designer might choose to include optional DirectShow or WDM kernel-mode streaming filter components or hardware that can provide the following capabilities:

- Greater than 85-dB dynamic range codec audio quality to the performance requirements of the consumer-electronics market
- Software or hardware Dolby AC-3 or MPEG-2 multichannel decode and downmix to stereo at 16-bit 48 kHz
- Software or hardware MPEG-1 layer-2 stereo at 16-bit 32, 44.1, or 48 kHz
- Software or hardware support for up to 24-bit 96-kHz linear PCM (LPCM) data, down-converted to 16-bit 48 kHz.

The following table presents actual measurements collected by IAL for software Dolby AC-3 decode on Intel Pentium II processors with MMX technology.

Performance Measurements for Dolby AC-3 Decode

Function	233-MHz Pentium II processor[1]
Dolby-certified 5.1-channel AC-3 decode and downmix to ProLogic-encoded stereo at 48 kHz	8%
Optional Dolby-certified virtual surround for rear channels, two channels at 48 kHz	12%[2]
Total for AC-3 decode with HRTF 3-D virtual surround	**20%**

[1] Actual performance is dependent on specific processor, chip-set, memory, and I/O subsystem implementations, as well as the software implementation (integer, FP, or MMX technology). To estimate the performance on other processor speeds with Intel MMX technology, use a ratio based on the processor's MHz rating.

[2] HRTF 3-D virtual surround performance is based on 3Q1997 release of Intel 3D RSX; performance of the 1Q1997 3D RSX is 24 percent.

Full-Duplex H.323/H.324 Video and Audio Conferencing

WDM audio supports the following features for full-duplex video and audio conferencing under Windows 98 and Windows NT 5.0:

- Native 32-bit DirectSound support for simultaneous audio input and output, not dependent on 16-bit MMSYS components
- Input and output position reporting mechanisms for synchronization of speaker and microphone streams, accurate to 1 ms or better
- WDM Stream class driver that provides access to acoustic echo cancellation (AEC) reference interfaces supported by hardware codecs
- WDM system audio filter graph that supports insertion of a non-Microsoft kernel-mode AEC filter module adjacent to the WDM Stream class driver in both the input and output streams

Baseline PC 98 audio hardware support for H.323/H.324 video and audio conferencing requires full-duplex audio capability. For a list of the PC 98 requirements, see the “Basic Audio Requirements” section later in this chapter.

The system designer might choose to include optional hardware to provide the following capabilities:

- Full-duplex operation at independent sample rates for simultaneous 8-kHz microphone input and 44.1-kHz speaker output
- Hardware AEC references for echo cancellation filtering

The following table presents actual measurements collected by IAL for H.323/H.324 audio on Intel Pentium II processors with MMX technology.

Performance Measurements for H.323/H.324 Audio

Function	233-MHz Pentium II processor[1]
Full-duplex G.723.1 encode and decode, mono at 8 kHz	10%
Full-duplex AEC, mono at 8 kHz	6%
Total H.323/H.324 audio	**16%**

[1] Actual performance is dependent on specific processor, chip-set, memory, and I/O subsystem implementations, as well as the software implementation (integer, FP, or MMX technology). To estimate the performance on other processor speeds with Intel MMX technology, use a ratio based on the processor’s MHz rating.

Basic Audio Requirements

This section defines basic PC 98 hardware feature requirements for audio components. These are system-based requirements, targeted for the entire PC solution as it ships, regardless of whether the audio components are separate add-on devices or are built into the system (for example, on the system board or the video monitor).

System Requirements for Audio

This section summarizes the PC 98 system requirements for audio.

1. PC system includes PC 98 audio capabilities

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Required*

Although audio is a standard feature in most PC market segments, it is understood that certain SOHO and Office PC designs that focus heavily on cost will not require audio. For PCs that include audio, the requirements defined in this chapter must be met.

2. Audio device does not connect to ISA bus

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required*	*Required*

In 1998, the transition away from ISA will be underway. Some of the reasons that ISA is not acceptable include the high overhead of transferring audio samples, the excessive requirements of resources such as IRQs, DMA, and I/O, plus limited power management capabilities.

There will be a delayed phase-in period for this requirement, during which only system-board ISA devices will be allowed. Please notice that the ISA requirements that apply during the transition phase are the same as those defined in *PC 97 Hardware Design Guide*, and are also summarized in the “Legacy Support” appendix in the References part of this guide.

Audio Performance and Feature Requirements

This section summarizes the required performance and features for audio on PC 98 systems.

Several companies have joined together to develop a standardized testing procedure for measuring audio performance titled “Personal Computer Audio Quality Measurement Definitions,” available on the web site at http://www.crystal.com/new/papers/meas.htm. Contributors to this specification include Audio Precision, Compaq Computer Corporation, Crystal Semiconductor, Intel Corporation, and Microsoft Corporation. Microsoft test procedures are based on this methodology and are available at http://www.microsoft/com/hwtest/.

3. Audio performance meets PC 98 requirements
Required

The following table summarizes audio performance requirements for all audio-enabled PC 98 systems, with the exceptions noted for mobile audio.

The tests isolate half-duplex play or record performance. Additional attention should be paid to full-duplex systems with an embedded microphone and speakers, such as mobile PCs and multimedia monitors, where acoustic coupling can significantly degrade microphone performance.

For precise definitions of the terminology used in the following charts, please refer to the test methodology paper at http://www.microsoft/com/hwdev/devdes/, as described earlier in this section.

PC 98 Audio Performance Requirements

Feature	Requirement	
Playback performance (PC-D-A) for line output	Frequency response (-3 dB)	
	44.1-kHz source material	20 Hz to 17.6 kHz[4]
	48.0-kHz source material	20 Hz to 19.2 kHz[4]
	Dynamic range (SNR)	≥80 dB FS A[1,2]
	THD+N	≤–60 dB FS A[3]
Recording performance (A-D-PC) for line input	Frequency response	
	44.1-kHz destination material	20 Hz to 17.6 kHz[4]
	48.0-kHz destination material	20 Hz to 19.2 kHz[4]
	Dynamic range (SNR)	≥75 dB FS A[2]
	THD+N	≤–55 dB FS A[3]
Analog pass-through performance (A-A)	Line input to line output	
	Frequency response (-3 dB)	20 Hz to 20.0 kHz[4]
	Dynamic range (SNR)	≥80 dB FS A[2]
	THD+N	≤–60 dB FS A[3]
	Line input to speaker output with 8-ohm load	
	Frequency response (-3 dB)	20 Hz to 20.0 kHz[4]
	Dynamic range (SNR)	≥70 dB FS A[2]
	THD+N	≤–50 dB FS A[3]
	Microphone input to line output	
	Frequency response (-3 dB)	100 Hz to 12.0 kHz
	Dynamic range (SNR)	≥70 dB FS A[2]
	THD+N	≤–50 dB FS A[3]
Full-scale input voltage	FSIP (A-D-PC) line input	1.0 to 2.0 Vrms
	FSIP (A-D-PC) microphone input	50 to 200 mVrms
Full-scale output voltage	FSOP (PC-D-A)	1.0 to 2.0 Vrms
Line output cross-talk	Channel separation between output channels (for example, left to right, center to right)	≥55 dB
Sampling frequency accuracy	Playback	0.1%
	Record	0.1%

[1] Decibels relative to full scale (FS), measured using “A weighting” filters.

[2] For mobile PCs, the dynamic range requirements are relaxed by10 dB FS A.

[3] For mobile PCs, the THD+N requirements are relaxed by 10 dB FS A.

[4] For mobile PCs, the required frequency response is 20Hz to 15kHz, measured using 6 dB corners.

4. Audio system provides support for basic data formats

Required

The audio system must provide full-duplex support of the following audio formats:

- Mono/stereo
- 8-bit unsigned
- 16-bit signed

The audio system must provide support for both recording and playing back of sample rates that include 8, 11.025, 16, 22.05, 44.1, and 48 kHz.

5. Audio system reports sample position for stream synchronization

Required

The driver must be capable of returning an accurate indication of the current position of the buffer being rendered. Sample accuracy is in relation to the samples given to the codec. Accurate reporting of the current position in the buffer being rendered means reporting the position within 1 ms. This requirement applies for both compressed and uncompressed data.

For information about WDM device driver support for streaming capabilities, see the Windows NT 5.0 DDK. See also the related articles available on the web at http://www.microsoft.com/hwdev/pcfuture/.

6. Audio system provides sufficient externally accessible inputs and outputs

Required

At a minimum, the audio system must have the following features:

- A monaural microphone, stereo line input, or both
- Stereo line-level output

These inputs and outputs can be located anywhere on the system, including on the speakers and video monitor.

7. Audio system connectors are labeled with icons as defined for PC 98

Required

To reduce user confusion, the external connections must use a consistent set of symbols based on the standard artwork defined in the “Icons” appendix in the References part of this guide.

Advanced Audio Recommendations

This section outlines the future direction of PC audio. It offers a view of the market requirements to come for PC audio, describing a number of directions that the industry is taking.

8. Audio performance meets PC 98 advanced recommendations

Recommended

The expectations for audio quality in the consumer-electronics industry are quite high. For those applications where audio quality is important, such as for Entertainment PC, it is strongly recommended that audio performance meet the capabilities recommended in the following table.

PC 98 Audio Performance Recommendations

Feature	Recommendation	
Playback performance (PC-D-A) for line output	Frequency response (-3 dB)	
	44.1-kHz source material	20 Hz to 20 kHz
	48.0-kHz source material	20 Hz to 20 kHz
	Dynamic range (SNR)	≥85 dB FS A
	THD+N	≤–65 dB FS A
Recording performance (A-D-PC) for line input	Frequency response (-3 dB)	
	44.1-kHz destination material	20 Hz to 20 kHz
	48.0-kHz destination material	20 Hz to 20 kHz
	Dynamic range (SNR)	≥80 dB FS A
	THD+N	≤–60 dB FS A
Analog pass-through performance (A-A)	Line input to line output	
	Frequency response (-3 dB)	20 Hz to 20.0 kHz
	Dynamic range (SNR)	≥85 dB FS A
	THD+N	≤–65 dB FS A
	Line input to speaker output	
	with 8-ohm load	20 Hz to 20.0 kHz
	Frequency response (-3 dB)	≥75 dB FS A
	Dynamic range (SNR)	≤–55 dB FS A
	THD+N	
	Microphone input to line output	
	with 8-ohm load	20 Hz to 20.0 kHz
	Frequency response (-3 dB)	≥75 dB FS
	Dynamic range (SNR)	≤–55 dB FS
	THD+N	

Continued

PC 98 Audio Performance Recommendations *(continued)*

Feature	Recommendation	
Full-scale input voltage	FSIP (A-D-PC) line input	2.0 to 3.0 Vrms
	FSIP (A-D-PC) microphone input	50 to 200 mVrms
Full-scale output voltage	FSOP (PC-D-A)	1.0 to 2.0 Vrms
Cross talk between signal channels	Stereo separation	≥65 dB A
Sampling frequency accuracy	Playback	0.1%
	Record	0.1%

9. Audio system supports full-duplex operation at independent sampling rates

Recommended

Voice recognition and audio/video conferencing require the audio system to simultaneously play back and record. Incoming and outgoing audio should be capable of operating at independent sampling rates. This recommendation considers the entire system, including the possibility of USB speakers or microphones.

10. Audio system provides hardware or software support for the Downloadable Samples specification

Recommended

Support for DLS as defined by the MIDI Manufacturers Association is recommended. For more information, see *DLS Specification, Version 1.0* or higher, at http://www.midi.org.

11. Audio system supports AEC reference inputs

Recommended

Full-duplex internal or external audio codecs that introduce additional digital or analog audio sources into the final mix are recommended to support simultaneous capture of microphone and AEC reference inputs. One analog-to-digital converter (ADC) is used to capture the microphone input, and another ADC is used to capture a monophonic representation of final output mix, which includes all digital and analog sources.

It is possible to use a single stereo ADC to capture the two monaural streams. The AEC reference should be time-synchronized and available at the same sample rate as the microphone input.

For more information, see Section 6.2 of the Audio Codec ’97 specification from Intel Corporation, which describes one possible implementation.

12. Audio system provides hardware filtering of HRTF 3-D filters

Optional

For those solutions that provide hardware acceleration of 3-D filters, support for eight separate audio sources is suggested, especially for Entertainment PCs.

13. CD, DVD, and broadcast audio playback meet PC 98 requirements

Required with DVD Video

These recommendations are specified to ensure quality playback of MPEG-2 audio from any source, including DVD, digital broadcast or satellite systems, hard drives, and so on. The goal for DVD and other A/V playback is to ensure that the end-user experience is the same or better than from a stand-alone DVD player.

For those PCs that support software or hardware decoding and playback of DVD-Video or MPEG-2 video, the audio decoder must be capable of supporting the following formats:

- Support for one or both of the following formats is required, depending upon the local requirements for DVD audio:
  - AC-3 (Dolby Digital) less than or equal to 5.1 channels, at 48 kHz less than or equal to 384 Kbps.
  - MPEG-2 multichannel less than or equal to 7.1 channels, at 48 kHz less than or equal to 912 Kbps.
- MPEG-1 Layer 2 stereo, at 44.1 and 48 kHz less than or equal to 384 Kbps.
- LPCM less than or equal to 8 channels, 16-bit, 20-bit, and 24-bit at 48 or 96 kHz less than or equal to 6.144 Mb/s.

Note: Conversion to 48-kHz 16-bit stereo is acceptable when the content exceeds the available resolution, sampling rates, or number of output channels.

14. Audio system provides consistent volume levels for different devices

Optional

In cases where each audio channel is set to the same position on the Windows mixer panel, it is suggested that each channel provide a comparable volume level. Users should not need to have radically different settings on the control panel to balance the relative volume of each audio channel.

PC 98 Design for Audio

This section summarizes requirements related to the PC 98 design initiatives as defined in Part 1 of this guide.

Plug and Play for Audio

The items in this section are requirements for all audio components.

15. Each device has a unique Plug and Play device ID

Required

Each bus-specific device must have a Plug and Play device ID for the bus it uses, as defined in Part 3 of this guide. For example, a PCI device must comply with PCI 2.1 requirements and also provide a Subsystem ID and Subsystem Vendor ID as defined in the “PCI” chapter in Part 3 of this guide. As another example, a USB device must provide a unique ID as defined in the *Universal Serial Bus Specification, Version 1.0* or higher.

Note: Each separate device or function enumerated by the BIOS on an audio adapter must have a separate Plug and Play device ID and separate resource configuration. If a game port or CD-ROM interface is supplied, resources must be allocated in addition to those required for the audio device. Such devices must also have independent dynamic disable capabilities. For complete information about requirements for multifunction cards, see the “Basic PC 98” chapter in Part 2 of this guide.

16. Automatic resource assignment and dynamic disable capabilities are supported

Required

The system must be capable of automatically assigning, disabling, and relocating the resources used by this device when necessary, using the method required for the related bus class. All configuration settings must be capable of being made through software, with no system reboot required.

When the end user changes this device or adds it to the system, setting resource assignments must not require changing jumpers or switches on either the adapter or the system board. In the event of an irreconcilable conflict with other devices on the system, the system must be able to disable the device in order to prevent the system from stalling. The device must not claim any resources while disabled.

Bus Design for Audio

This section defines the requirements for bus-specific design for PC 98 audio.

Requirements for PCI Audio Devices

For audio devices that connect to the PCI bus, the following requirements apply.

17. PCI device conforms to PCI 2.1 and additional PC 98 requirements
Required

If the device uses PCI, it must meet the requirements defined in the “PCI” chapter in Part 3 of this guide, including requirements for providing a Subsystem ID and Subsystem Vendor ID and for complying with PCI 2.1.

18. PCI device supports initiator, target, and block transfer
Required

For complete implementation details, see PCI 2.1.

Full-duplex audio sample transport must be supported using separate PCI bus mastering hardware for playback and capture sample streams.

It is desirable for sample transport mastering hardware to support burst capabilities in order to read or write multiple samples within the same PCI bus transaction. This will lessen the impact of sample transport on other agents in the system, which will have a positive effect on the system’s responsiveness.

19. PCI audio components use a suitable configuration scheme if using ISA resources
Required

If a PCI audio device requires the use of ISA resources such as IRQs or DMA, it must configure these resources in a way that meets the following objectives:

- Any resource enabled while Windows or an MS-DOS window is active must be reported to the Windows Device Manager. Conversely, if a PCI device uses ISA resources in MS-DOS mode that do not appear in the Configuration Manager, then the device must disable these resources before Windows restarts.
- The ISA resources allocated by a PCI device must be used only for MS-DOS mode or for applications running in an MS-DOS window. Windows-based applications must not require the use of ISA resources.

Two acceptable means for achieving these objectives are as follows:

- For a PCI audio device on the system board, the BIOS can configure the ISA resources and present a device node to the operating system. The operating system can then understand and reconfigure the ISA resources.
- For a PCI audio device present in an add-on card, the device does not use any ISA resources while running Windows or an MS-DOS window. Only when the user enters MS-DOS mode does the ISA resource become enabled.

PCI devices on the system board do not have to provide relocatable I/O addresses for the following legacy I/O registers:

- Sound Blaster (220h to 22Fh)
- FM synthesis (388h to 38Bh)
- MPU 401 (330h to 331h)
- Windows Sound System (WSS) compatible (534h to 537h)
- Game port (200 to 207h)

20. PCI device is digital ready

Required

In order to transfer digital audio to USB or IEEE 1394 devices, all digital audio data created in the PC must be available to the operating system for mixing and streaming. All PCI audio devices must be able to route the final mix of all digital audio data created or processed on-chip to the host using bus master transfers.

Support for capture and inclusion of internal analog resources in the final mix is desirable but not required. CD-ROM drives that support direct reading of Red Book data through the primary interface are strongly recommended.

For example, a PCI audio device provides HRTF 3-D filtering and wave-table synthesis. After mixing all of the separate 3-D sources and wave-table channels down to a single stereo stream, the device transfers the data to host memory.

Requirements for USB Audio Devices

For audio devices that connect to a USB port, the following requirement applies.

21. Audio meets USB specification and USB audio device class specification
Required

The device must comply with *Universal Serial Bus Specification, Version. 1.0* or higher, and the USB device class specification for audio. This ensures that all Plug and Play requirements are met and that drivers provided with the operating system support this device.

Requirements for IEEE 1394 Audio Devices

For audio devices that connect to the IEEE 1394 bus, the following requirement applies.

22. Audio meets PC 98 requirements for IEEE 1394
Required

For requirements related to IEEE 1394 peripherals, see the “IEEE 1394” chapter in Part 3 of this guide.

Power Management for Audio

This section summarizes the power management requirements for audio components.

23. System and device comply with PCI bus power management specification
Required

PCI-based audio controllers must comply with the *PCI Bus Power Management Specification, Revision 1.0* or higher (PCI PM). Any PCI add-on audio device must comply with PCI PM. Audio devices implemented on the system board must comply fully with the ACPI 1.0 specification.

24. Audio device complies with device class power management reference specification
Required

The *Audio Device Class Power Management Reference Specification, Version 1.0* or higher, provides definitions of the OnNow device power states (D0–D3) for these devices. The specification also covers device functionality expected in each power state and the possible wake-up event definitions for the class. Implementation of power states D0 and D3 is required. Other power states are optional.

25. Device supports wake-up events

Optional

For PC 98, the ability to cause a wake-up event as defined in the *Audio Device Class Power Management Reference Specification, Version 1.0* or higher, is an optional feature.

Device Drivers and Installation for Audio

This section summarizes requirements for audio device drivers.

26. Device drivers and installation meet PC 98 requirements

Required

The manufacturer does not need to supply a driver for a device if the device passes PC 98 compliance testing using a driver provided with the operating system. If the manufacturer supplies the drivers, the requirements for the device drivers and installation are defined in the “Basic PC 98” chapter in Part 2 of this guide. The basic requirements include driver support for unattended installation and Help file support if special driver parameters are used.

27. Audio meets PC 98 requirements for WDM driver support

Required

All audio devices must have drivers that use the 32-bit WDM architecture exclusively. The manufacturer can either supply a WDM driver with the audio device or rely on a WDM driver provided with Windows and Windows NT. For information, see the Windows NT 5.0 DDK.

28. Applications provided with device meet Win32 requirements

Required

Any Windows-based applications provided with the device must meet software compatibility requirements as defined in the Win32 SDK.

Audio References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

Advanced Configuration and Power Interface Specification, Revision 1.0
http://www.teleport.com/~acpi/

Audio '98 Roadmap, Audio Codec '97 Component Specification, and
Audio Codec '97 Design Guide papers
Intel Corporation
http://developer.intel.com/pc-supp/platform/aud98/

Audio design for Windows operating systems white papers
http://www.microsoft.com/hwdev/devdes/

Audio Device Class Power Management Reference Specification, Version 1.0
http://www.microsoft.com/hwdev/onnow.htm

Downloadable Samples (DLS) specification
MIDI Manufacturers Association
Fax: (714) 736-9775
E-mail: mma@midi.org
http://www.midi.org

Personal Computer Audio Quality Measurement Definitions
by Dr. Steven Harris and Cliff Sanchez, Crystal Semiconductor
http://www.crystal.com/new/papers/meas.htm

PCI Local Bus Specification, Revision 2.1 (PCI 2.1)
PCI Bus Power Management Specification, Revision 1.0 (PCI PM)
http://www.pcisig.com

Plug and Play specifications
http://www.microsoft.com/hwdev/specs/

Universal Bus Specification, Version 1.0
USB device class specifications
http://www.usb.org

WDM device driver support white papers
http://www.microsoft.com/hwdev/pcfuture/

Windows DDK, Windows NT DDK, DirectX DDK, and Win32 SDK
MSDN Professional membership

Checklist for Audio Components

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

Consumer PC 98	Office PC 98	Entertainment PC 98
1. PC system includes PC 98 audio capabilities		
Recommended	*Recommended*	*Required*
2. Audio device does not connect to ISA bus		
Required	*Required*	*Required*
3. Audio performance meets PC 98 requirements		
Required		
4. Audio system provides support for basic data formats		
Required		
5. Audio system reports sample position for stream synchronization		
Required		
6. Audio system provides sufficient externally accessible inputs and outputs		
Required		
7. Audio system connectors are labeled with icons as defined for PC 98		
Required		
8. Audio performance meets PC 98 advanced recommendations		
Recommended		
9. Audio system supports full-duplex operation at independent sampling rates		
Recommended		
10. Audio system provides hardware or software support for the Downloadable Samples specification		
Recommended		
11. Audio system supports AEC reference inputs		
Recommended		
12. Audio system provides hardware filtering of HRTF 3-D filters		
Optional		
13. CD, DVD, and broadcast audio playback meet PC 98 requirements		
Required with DVD Video		
14. Audio system provides consistent volume levels for different devices		
Optional		
15. Each device has a unique Plug and Play device ID		
Required		
16. Automatic resource assignment and dynamic disable capabilities are supported		
Required		
17. PCI device conforms to PCI 2.1 and additional PC 98 requirements		
Required		

18. PCI device supports initiator, target, and block transfer
Required

19. PCI audio components use a suitable configuration scheme if using ISA resources
Required

20. PCI device is digital ready
Required

21. Audio meets USB specification and USB audio device class specification
Required

22. Audio meets PC 98 requirements for IEEE 1394
Required

23. System and device comply with PCI bus power management specification
Required

24. Audio device complies with device class power management reference specification
Required

25. Device supports wake-up events
Optional

26. Device drivers and installation meet PC 98 requirements
Required

27. Audio meets PC 98 requirements for WDM driver support
Required

28. Applications provided with device meet Win32 requirements
Required

CHAPTER 18

Storage and Related Peripherals

This section presents the PC 98 requirements for storage and related peripherals, including DVD devices. Specific requirements for SCSI, IDE, and ATAPI peripherals are defined in the related chapters in Part 3 of this guide.

For specific information about implementation details related to storage devices under the Windows and Windows NT operating systems, see the articles at http://www.microsoft.com/hwdev/devdes/.

Contents

Storage Peripherals Basic Features

This section summarizes the hardware requirements for storage peripherals. For related acoustical requirements for storage devices, see the “Basic PC 98” chapter in Part 2 of this guide.

1. Storage device and controller support bus master capabilities

Required

Bus master capabilities must meet the related specification for the particular controller. For example, the programming register set for PCI IDE bus master DMA is defined in SFF 8038i.

Correctly implemented bus master support ensures improved performance and Windows-compatible device driver support.

Note: This requirement does not apply to legacy FDCs and will not become a requirement for legacy FDCs.

2. Removable media includes media status notification support

Required

The following list shows the required specifications for implementing media status notification, depending in device type.

Device type	Media status notification implementation
CD-ROM and DVD-ROM	Required. Comply with all provisions in the Media Event Status Notification subsection of SFF 8090 (Mt. Fuji specification), available at ftp://fission.dt.wdc.com/pub/standards/SFF/specs/.
Other types of IDE/ATAPI removable storage devices	Required. Comply with *Media Status Notification Support, Version 1.03* or higher, available at http://www.microsoft.com/hwdev/specs/.
ATAPI floppy drives	Required. Comply with media status notification as defined in SFF 8070.
Other ATAPI devices, including tape drives	Recommended. Comply with SFF 8090 (Mt. Fuji specification) if implemented.
Other types of SCSI removable devices	Not required. Comply with SFF 8090 (Mt. Fuji specification) if implemented.

3. Option ROMs support Int 13h Extensions

Required

The Int 13h Extensions ensure correct support for high-capacity drives, consistent drive-letter mapping between real mode and protected mode, and other capabilities for both Windows and Windows NT. Support for the fixed-disk access subset of Int 13h Extensions must be provided in the system BIOS and in any option ROMs for storage devices that include BIOS support. The Int 13h Extensions are defined in the “Layered Block Device Drivers” section of the Windows 98 DDK and in the Windows NT 5.0 DDK.

4. Device meets PC 98 general device requirements

Required

These include the PC 98 requirements for a Plug and Play device ID, automated software-only settings for device configuration, device drivers and Windows-based installation, and icons for external connectors. For more information, see the “Basic PC 98” chapter in Part 2 of this guide.

5. Device meets PC 98 requirements for ports or buses

Required

The device must meet all requirements for the port or bus to which it is attached. For example, a drive that uses the parallel port must meet all the requirements defined for legacy parallel peripherals (including requirements for ECP mode), as defined in the “I/O Ports and Devices” chapter in Part 4 of this guide. If the device uses a PCI, IDE, or SCSI connection, the device must meet the related requirements defined in Part 3 of this guide.

6. Device Bay storage device meets PC 98 requirements

Required

All Device Bay peripherals must meet the requirements defined in *Device Bay Interface Specification, Version 1.0.* Any storage device designed as a Device Bay peripheral must also interface with either the USB or IEEE 1394 bus, or both, and must support the *Universal Serial Bus Device Class Definition for Mass Storage Devices, Version 1.0* or higher, if it interfaces with USB.

7. IDE/ATAPI devices supported on IEEE 1394

Recommended

For PC 98 systems, IDE/ATPI device support on IEEE 1394 is recommended. An interface device or dongle that allows all IDE/ATAPI devices to be connected to IEEE 1394 can be included with the system. If IDE/ATAPI support is included, the operating system must be able to boot from this device.

This recommended support is defined in relation to the emerging Device Bay standard. For more information about PC 98 recommendations and requirements for Device Bay, see the “Basic PC 98” chapter in Part 2 of this guide.

8. IDE/ATAPI devices and controllers support Ultra DMA/33
Required
Ultra DMA/33 (also known as Ultra-ATA) is required to avoid the bottleneck created by the current 16.6 Mb/s limit on disk transfer. Ultra DMA/33 also provides error checking for improved robustness over previous IDE implementations. This is a requirement for all IDE/ATAPI controllers and devices.

PCI chip sets must implement DMA as defined in SFF 8020i and must implement Ultra DMA/33 as defined in the specification submitted by Quantum Corporation for inclusion in the ATA-4 specification.

9. USB-based mass storage device meets PC 98 requirements for USB
Required
If a USB-based mass-storage device (including tape and CD-ROM) is implemented in a PC 98 system, it must meet the requirements defined in the "USB" chapter in Part 3 of this guide. It must also meet the requirements defined in *Universal Serial Bus Device Class Definition for Mass Storage Devices, Version 1.0* or higher.

10. System BIOS or option ROM supports El Torito No Emulation mode
Required
For PC systems that include CD-ROM or DVD-ROM devices, the system BIOS or option ROM must support the No Emulation mode defined in the specification *El Torito—Bootable CD-ROM Format Specification, Version 1.0,* published by IBM and Phoenix.

Floppy Disk Controller

This section describes the specific requirements for any FDC provided with a PC 98 system. The device must also meet the general requirements defined in the "Storage Peripherals Basic Features" section earlier in this chapter and the "PC 98 Design for Storage Components" section later in this chapter.

PC 98 does not require an FDC. Although most systems include some form of floppy disk drive, some Office PC 98 systems might not need one.

11. Floppy disk capabilities provided through expansion card or external bus
Recommended
To support migration away from legacy devices, it is recommended that support for floppy disk drives be provided by a solution based on an external bus, such as USB or PC Card, or an expansion card for SCSI or IDE.

12. IDE floppy drive complies with SFF 8070

Required

The SFF 8070 specification defines the required implementation supported by the Windows operating system, including support for ATA floptical drives.

13. Legacy FDC built into system

Optional

Including a legacy FDC on a PC 98 system is optional. However, if a legacy FDC is included, it must meet the requirements listed in this section.

14. Legacy FDC device meets resource configuration requirements

Required

The following resource requirements must be met for each FDC device on the system (not shared among devices of the same type):

- Use static I/O addresses 3F2h, 3F4h, and 3F5h. Additional addresses can be provided in the event of conflict.
- Use IRQ 6.
- Use DMA Channel 2 if FDC supports block data transfers to memory using DMA controllers.

15. System supports conflict resolution and dynamic disable capabilities for legacy FDC

Required

The FDC must be capable of being disabled. For example, if the FDC is located on the system board and an adapter card that includes an FDC is added to the system, the system-board FDC must be capable of being disabled to prevent conflicts with the new card. If the FDC is located on an expansion card, the expansion card must allow independent dynamic disabling of the FDC and the hard disk controller. In this case, the adapter will continue to function if the FDC is disabled because of conflicts, and vice versa.

Hard Disk Drives

This section summarizes specific requirements for hard disk drives. The device must also meet the general requirements defined in the “Storage Peripherals Basic Features” and “PC 98 Design for Storage Components” sections in this chapter.

Note: In the “IDE and ATAPI” chapter in Part 3 of this guide, BIOS support is required for an LBA scheme compatible with the BIOS/CMOS and IDE register-set constraints. This enables support for IDE disk drives larger than 528 MB.

16. IDE hard drive is SMART-compliant and uses SMART IOCTL API
Required

The Self-Monitoring, Analysis, and Reporting Technology system (SMART) is an industry term used to describe technology that monitors and predicts device performance.

The *SMART IOCTL API Specification, Version 1.1* or higher, published by Compaq Computer Corporation and Microsoft Corporation, describes the API used by an application to issue SMART commands to an IDE drive under Microsoft Windows 98. In Windows, the API is implemented in a vendor-specific driver: Smartvsd.vxd.

For all PC 98 systems, the hard drive must be SMART-compliant.

17. IDE hard drive spin-up time supports OnNow capabilities
Recommended

The hard disk drive should spin up and be able to complete a Read operation within 6 seconds of applying power and within 5 seconds of leaving the ATA STANDBY mode and transitioning to ATA ACTIVE, as specified in *Storage Device Class Power Management Reference Specification, Version 1.0* or higher.

This recommendation supports the OnNow design initiative goals for a system that is “instantly” available when power is applied.

CD-ROM Peripherals

This section summarizes the requirements for CD-ROM peripherals and the specific features for PC 98. The device must also meet the general requirements defined in the “Storage Peripherals Basic Features” and “PC 98 Design for Storage Components” sections in this chapter, including the requirement for bus mastering.

18. CD-ROM drive provides 8x or higher performance
Required

The CD-ROM drive must support 1200K per second average throughput or higher performance when running in the fully on power state.

19. CD-ROM drive is CD-Enhanced compatible
Required

For PC 98, the CD-ROM drive must be able to mount multisession CD-ROM discs, even if track 1 is Red Book audio. Microsoft recommends use of the Sony ReadTOC method for SCSI-2 multisession support as noted in SFF 8020i, Version 2.5.

CD-Enhanced support must be Blue Book compliant, as defined in *Enhanced Music CD Specification, Version 1.0.*

20. CD-ROM drive supports specified logical and physical CD formats

Required

At a minimum, the CD-ROM device must be compatible with the following formats to ensure cross-media compatibility:

- Logical formats: CD Red Book, Yellow Book, White Book, and Blue Book
- Physical formats: CD-ROM and CD-Audio

CD-E and CD-R (Orange Book format) can be implemented; however, there are no PC 98 requirements or recommendations for this format.

Note: Any ATAPI CD-ROM drive designed to play back CD-I content must return a minimum of two track entries for the READ_TOC (0x43) command. These two track entries must be a track 01 entry and a track 0xAA entry for the lead-out address. Drives that do not comply with this minimum requirement cannot play back CD-I movies.

21. IDE/ATAPI CD-ROM drive complies with SFF 8020i, Version 1.2

Required

CD-ROM drives attached to the system must support the hardware and protocols documented in *ATA Packet Interface for CD-ROMs, Version 1.2* or higher.

Note: For PC 98, support for the READ CD-DA command as defined in SFF 8020i, Version 1.2, is recommended. This might become a requirement in future versions of these guidelines.

For DVD drives, see the “DVD device meets SFF 8090 specification” requirement later in this chapter.

22. CD-ROM drive supports multisession and compatibility forms of the READ_TOC command

Required

Both multisession forms (01b and 10b) as well as the compatibility form (00b) of the READ_TOC command must be implemented. This ensures complete support for CD-ROM multisession capabilities.

Note: Any ATAPI CD-ROM drive designed to play back CD-I content must return a minimum of two track entries for the READ_TOC (0x43) command. These two track entries must be a track 01 entry and a track 0xAA entry for the lead-out address. Drives that do not comply with this minimum requirement cannot play back CD-I movies.

23. IDE/ATAPI CD changer meets SFF 8070 specification

Required

If an ATAPI-compatible CD changer is present that has a capacity for seven or fewer discs, the device must comply with SFF 8070.

Rewritable ATAPI Devices

This section summarizes specific requirements for optical storage devices. The device must also meet the general requirements defined in the “Storage Peripherals Basic Features” and “PC 98 Design for Storage Components” sections in this chapter.

24. ATAPI rewritable device meets SFF 8070i specification
Required
The SFF 8070i specification defines the requirements for ATAPI rewritable devices, including specifications for logical unit number (LUN) implementation, media status notification, and device write protection. This also includes required support for the Read Format Capacities command.

DVD Devices

This section summarizes specific requirements for DVD devices. The device also must meet the general requirements defined in the “Storage Peripherals Basic Features” and “PC 98 Design for Storage Components” sections in this chapter.

For information about the PC 98 requirements for DVD-Video and MPEG-2 playback performance, see the “Video and Broadcast Components” chapter in Part 4 of this guide.

For more information about DVD support under Windows and Windows NT, see the articles at http://www.microsoft.com/hwdev/devdes/.

25. DVD drive supports bus master DMA transfers
Required
Hardware decoders must support byte-aligned, multisegment, bus master DMA transfers. The drive must function without corrupting data in DMA mode.

ATAPI DVD drives and IDE system-board implementations must support DMA as specified in SFF 8090 (Mt. Fuji specification). DMA must be enabled by default.

26. DVD drive meets minimum compatibility requirements
Required
At a minimum, the DVD device must be compatible with the following formats to ensure that the DVD device can read earlier media:

- Logical formats: CD Red Book, Yellow Book, White Book, and Blue Book
- Physical formats: CD-ROM, CD-Audio, DVD-ROM, and DVD-RAM 1.0

The device must also be able to mount multisession CD-ROM discs, as described in the “CD-ROM drive is CD-Enhanced–compatible” requirement earlier in this chapter. However, there is no DVD drive support for CD-E and CD-R.

27. Device and driver support DVD command sets

Required

The device and driver must support the command set defined in SFF 8090 (Mt. Fuji specification). Specifically, the device and driver must support the commands in the following list.

Code	Command name	Code	Command name
12h	Inquiry	42h	Read sub-channel
00h	Test unit ready	Beh	Read CD
03h	Request sense	B9h	Read CD MSF
55h	Mode select (10)	45h	Play audio (10)
5Ah	Mode sense (10)	47h	Play audio MSF
BDh	Mechanism status	4Bh	Pause/resume
25h	Read C/DVD capacity	4Eh	Stop play/scan
23h	Read formatted capacities	BAh	Scan
Adh	Read DVD structure	28h	Read (10)
A8h	Read (12)	08h	ATAPI soft reset
A7h	Set read ahead	E5h	Check power mode
1Bh	Start/stop unit	90h	Execute drive diagnostic
1Eh	Prevent/allow medium removal	E1h	Idle immediately
2Bh	Seek	00h	NOP
4Ah	Get event status notification	A0h	ATAPI packet
A4h	Report key	A1h	ATAPI identify device
A3h	Send key	Efh	Set features
43h	Read TOC	E6h	Sleep
44h	Read header	E0h	Standby immediate

28. DVD device meets SFF 8090 specification

Required

SFF 8090 (Mt. Fuji specification) defines the implementation requirements that the Windows operating system supports. For PC 98, a DVD device must comply with the following portions of SFF 8090:

- Media Event Status Notification
- Power management
- DMA implementation

29. DVD device uses high-speed expansion bus

Required

The DVD hardware must use a bus that supports high-speed transfer of multiple data types. Any DVD controller must be capable of sustained rates of 12 Mb/s minimum.

30. DVD drive supports UDF

Required

The drive must support UDF as defined in *Universal Disk Format Specification, Version 1.02* or higher, available from the Optical Storage Technology Association at http://www.osta.org.

31. DVD device uses push-to-close design

Recommended

A motorized design is not required, but if it is implemented, the device must be designed so the user has three options for closing the device when inserting a disc:

- Physically pushing on the bay.
- Physically pushing the close button on the bay housing.
- Selecting a software-supported option to close the device.

32. DVD device supports defect management

Required

The drive must support defect management that is transparent to the operating system, according to industry standards. Defect management is defined in *DVD Specification, Book A: Physical Specifications,* published by Toshiba Corporation.

33. DVD device supports copyright protection

Required

The drive must support a licensed CSS copyright-protection scheme and support CSS-protected discs to ensure proper protection for all content produced in accordance with CSS, as defined in the DVD specification.

Software is provided as part of the Windows and Windows NT operating system support for DVD in order to facilitate the authentication process required by this scheme. This allows a DVD-ROM drive to authenticate and transfer keys with a CSS decrypter. Windows and Windows NT operating system software will act as the agent to allow either hardware or software decrypters to be authenticated.

For information, see the related articles on DVD support under Windows and Windows NT at http://www.microsoft.com/hwdev/devdes/.

PC 98 Design for Storage Components

This section summarizes requirements related to Plug and Play and other bus-related and resource-related design issues for storage devices.

Plug and Play and Bus Design for Storage Components

The items in this section are requirements for Plug and Play capabilities.

34. Each device has a Plug and Play device ID

Required

For each device, there must be a device-specific ID. Each device must provide Plug and Play device IDs in the manner required for the bus it uses as defined in Part 3 of this guide. For example, a PCI add-on device must comply with PCI 2.1 requirements and also must provide a Subsystem ID and Subsystem Vendor ID, as defined in the "PCI" chapter in Part 3 of this guide.

35. Conflict resolution and dynamic disable capabilities supported for all devices

Required

To ensure conflict resolution for resource allocation, the device must conform to the Plug and Play specifications for the bus it uses as described in Part 3 of this guide. The system must have a method for automatically disabling or relocating the resources used by the device if conflicts occur when an expansion card is added to the system.

Devices must be capable of being disabled with software settings only, that is, without requiring rebooting or jumper setting changes. Disabling the device must result in freeing all its resources for use by other devices. DIP switches on boot devices can be used for an initial power-on default state or for non-Plug and Play system compatibility, but must be able to be overridden by software configuration after system power up.

The primary hard disk controller is not required to support dynamic disable capabilities.

Note: This requirement does not apply to jumper settings used by the OEM to make basic system-related settings in the factory. This requirement applies only to settings that the end user must make to configure the hardware.

36. 3F7h and 377h are unclaimed by devices

Required

To avoid having two devices in the system claim 3F7h and 377h, these addresses should not be claimed as resources in device registers, especially by IDE devices.

It is recognized that some FDC devices claim this range. Such devices can be implemented in a PC 98 system; however, the system manufacturer must ensure that only a single device in the system claims this range.

37. Physical security is provided for storage devices

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Recommended*

External drive devices should have locking capabilities. This is recommended for PC 98 systems and required for Net PC hardware. Each removable media device should be capable of being locked to prevent unauthorized access to data. This means that the device is rendered useless, either electronically or mechanically.

Power Management for Storage Components

This section summarizes specific power management requirements for storage devices.

38. Device and controller comply with device class power management reference specification

Required

The *Storage Device Class Power Management Reference Specification, Version 1.0* or higher, provides definitions of the OnNow device power states (D0–D3) for these devices. The specification also covers device functionality expected in each power state and possible wake-up event definitions for the class. Power states D0, D1, and D3 are PC 98 requirements for hard disks, CD-ROM drives, and other mass storage devices. Support for the D1 state is not required for floppy disk devices.

39. Device supports wake-up events

Optional

For PC 98, the ability to cause a wake-up event as defined in the *Storage Device Class Power Management Reference Specification, Version 1.0* or higher, is an optional feature.

Device Drivers and Installation for Storage

This section summarizes the basic requirements for device drivers and installation procedures for storage devices.

40. Device drivers and installation meet PC 98 requirements

Required

The manufacturer does not need to supply a driver if a PC 98-compliant driver provided with the operating system can be used. If the manufacturer supplies a driver, the requirements for the device drivers and installation are defined in the "Basic PC 98" chapter in Part 2 of this guide. The basic requirements include driver support for unattended installation and Help file support if special driver parameters are used.

Ease-of-use requirements for installation and configuration are defined for SCSI peripherals and for IDE and ATAPI devices in Part 3 of this guide. For information about WDM support for devices that use a USB or IEEE 1394 bus, see the Windows NT 5.0 DDK. See also the related articles on the web site at http://www.microsoft.com/hwdev/pcfuture/.

41. Device and file system run in protected mode following installation

Required

The device driver and the file system must be running in 32-bit protected mode (not compatibility mode) immediately following installation.

Note: Although it is strongly preferred that a system reboot not be required as part of device installation, it is recognized that installation of boot devices presents a special situation. For PC 98, it is acceptable that installation includes restarting the system during installation of a boot device.

42. Applications provided with the device meet Win32 requirements

Required

Any Windows-based applications provided with the device must meet Microsoft requirements for software compatibility as defined in the Win32 SDK. However, any software applications included with the device can be installed using an alternate Windows-based installation method as defined in the Win32 SDK.

43. Driver for partitioned media supports all Windows and Windows NT partition types

Required

Device drivers that support partitioned media must support all Windows and Windows NT partition types, which include but are not limited to FAT16, FAT32, NTFS, and UDF.

44. Driver for block-mode device supports extended BPBs
Required

Storage subsystems that include an MS-DOS–based block-mode device driver (for example, Aspidisk.sys) must support Extended BIOS Parameter Blocks (BPBs) in the Build BPB device driver function call, and must support category=48 in the generic IOCTL device driver interface calls, as specified in the 1996 update to the Windows DDK.

Storage References and Resources

This section lists resources for building storage hardware that works with the Windows and Windows NT operating systems.

ATA-2 [X3T9.2 948D], *ATA Packet Interface for CD-ROMs,* SFF 8020i, and other SFF Committee publications
ftp://fission.dt.wdc.com/pub/standards/SFF/specs/

Device Bay Interface Specification, Version 1.0
http://www.device-bay.org

Device driver support for storage devices and DVD white papers
http://www.microsoft.com/hwdev/devdes/

DVD Specification, Book A: Physical Specifications, Toshiba Corporation.
http://www.toshiba.com

El Torito—Bootable CD-ROM Format Specification, Version 1.0
http://www.ptltd.com/techs/specs.html

FAT32 partition device driver support
http://www.microsoft.com/hwdev/devdes/

IDE and SCSI specifications
SFF Committee publications
FaxAccess: (408) 741-1600 (fax-back)
Fax: (408) 867-2115

Global Engineering Documents
Phone: (800) 854-7179 (US)
(613) 237-4250 (Canada)
(303) 792-2181 (Outside North America)
Fax: (303) 397-2740
ftp://ftp.symbios.com/pub/standards/io/

Media Status Notification Support Specification, Version 1.03
Plug and Play specifications
SMART IOCTL API Specification, Version 1.1
http://www.microsoft.com/hwdev/specs/

Multisession Compact Disc Specification
Enhanced Music CD Specification, Version 1.0
Philips Consumer Electronics B.V.
Coordination Office Optical–Magnetic Media Systems
Building SWA-109, PO Box 80002
5600 JB Eindhoven, The Netherlands
Fax: (31) (40) 732113

Sony/Philips CD-ROM hardware logo program:
Bert Gall, Philips Consumer Electronics
Philips Consumer Electronics B.V.

Storage Device Class Power Management Reference Specification, Version 1.0
http://www.microsoft.com/hwdev/onnow.htm

Universal Disk Format Specification, Version 1.02
http://www.osta.org

Universal Serial Bus Device Class Definition for Mass Storage Devices, Version 1.0
Phone: (503) 264-0590
Fax: (503) 693-7975
http://www.usb.org

WDM device driver support white papers
http://www.microsoft.com/hwdev/pcfuture/

Windows DDK, Windows NT DDK, and Win32 SDK
MSDN Professional membership
(The Windows DDK includes information about the Int 13h Extensions API.)

Checklist for Storage and Related Peripherals

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

Consumer PC 98	Office PC 98	Entertainment PC 98

1. Storage device and controller support bus master capabilities
Required

2. Removable media includes media status notification support
Required

3. Option ROMs support Int 13h Extensions
Required

4. Device meets PC 98 general device requirements
Required

5. Device meets PC 98 requirements for ports or buses
Required

6. Device Bay storage device meets PC 98 requirements
Required

7. IDE/ATAPI devices supported on IEEE 1394
Recommended

8. IDE/ATAPI devices and controllers support Ultra DMA/33
Required

9. USB-based mass storage device meets PC 98 requirements for USB
Required

10. System BIOS or option ROM supports El Torito No Emulation mode
Required

11. Floppy disk capabilities provided through expansion card or external bus
Recommended

12. IDE floppy drive complies with SFF 8070
Required

13. Legacy FDC built into system
Optional

14. Legacy FDC device meets resource configuration requirements
Required

15. System supports conflict resolution and dynamic disable capabilities for legacy FDC
Required

16. IDE hard drive is SMART-compliant and uses SMART IOCTL API
Required

17. IDE hard drive spin-up time supports OnNow capabilities
Recommended

18. CD-ROM drive provides 8x or higher performance
Required

Consumer PC 98	Office PC 98	Entertainment PC 98
19. CD-ROM drive is CD-Enhanced compatible		
Required		
20. CD-ROM drive supports specified logical and physical CD formats		
Required		
21. IDE/ATAPI CD-ROM drive complies with SFF 8020i, Version 1.2		
Required		
22. CD-ROM drive supports multisession and compatibility forms of the READ_TOC command		
Required		
23. IDE/ATAPI CD changer meets SFF 8070 specification		
Required		
24. ATAPI rewritable device meets SFF 8070i specification		
Required		
25. DVD drive supports bus master DMA transfers		
Required		
26. DVD drive meets minimum compatibility requirements		
Required		
27. Device and driver support DVD command sets		
Required		
28. DVD device meets SFF 8090 specification		
Required		
29. DVD device uses high-speed expansion bus		
Required		
30. DVD drive supports UDF		
Required		
31. DVD device uses push-to-close design		
Recommended		
32. DVD device supports defect management		
Required		
33. DVD device supports copyright protection		
Required		
34. Each device has a Plug and Play device ID		
Required		
35. Conflict resolution and dynamic disable capabilities supported for all devices		
Required		
36. 3F7h and 377h are unclaimed by devices		
Required		
37. Physical security is provided for storage devices		
Recommended	*Recommended*	*Recommended*
38. Device and controller comply with device class power management reference specification		
Required		

Consumer PC 98	Office PC 98	Entertainment PC 98

39. Device supports wake-up events
Optional

40. Device drivers and installation meet PC 98 requirements
Required

41. Device and file system run in protected mode following installation
Required

42. Applications provided with the device meet Win32 requirements
Required

43. Driver for partitioned media supports all Windows and Windows NT partition types
Required

44. Driver for block-mode device supports extended BPBs
Required

C H A P T E R 1 9

Modems

This chapter presents the PC 98 requirements for modems, fax modems, voice modems, voice/data modems, wireless and cellular modems, and Integrated Service Digital Network (ISDN) modems.

For communications that require a solution based on NDIS under Windows and Windows NT, see the “Network Communications” chapter in Part 4 of this guide.

Contents

PC 98 Modem Design Issues

This section presents the key design issues for modems. The Windows and Windows NT operating systems and Win32-based applications use data, fax, voice, and voice/data integration features in modems.

Note: Communications standards mentioned in this chapter are available through ITU Sales, Bellcore, ETSI, or Global Engineering Documents, as described in the "Modem References" section at the end of this chapter.

Basic PC 98 Modem. The fundamental design principle for compatibility with Windows is for the device to be supported by the Universal Modem Driver (Unimodem), which uses INF files to characterize the behavior of a device. A detailed definition of Unimodem requirements is contained in the MDK, available at ftp://ftp.microsoft.com/developr/drg/modem/modemdev.exe.

The new PC 98 requirements and recommendations are directed at the following issues:

- Migrating to even higher speeds of dial-up access based on pulse-code modulation (PCM) modems (that is, ITU V.pcm, X2, and K56Flex).
- Addressing persistent cost-of-ownership problems, particularly modem detection and installation and Internet Service Provider (ISP) call failures.
- Migrating modem functions into the operating system to save costs and to provide upgrade flexibility.
- Addressing the failure of many voice modems to meet the needs of ISVs who are implementing computer-based telephony and need new features to be successful.

Migrating to Higher Speeds with PCM Modems. V.34, as advanced as it is, is not the end of the line for telephone-line modems. ISP access is the driving force for modem use. ISPs use high-density modem pools, which connect to the Public Switched Telephone Network (PSTN) using T1 or ISDN primary rate interface (PRI) links. This creates an opportunity, because the ISP-side modem has direct access to the inside of the digital telephone network, so it can directly modulate the PCM B-channel data.

PCM means that the impairments on the ISP-side analog loop have disappeared; there is no quantization noise or echo. The symbol rate on the remaining loop jumps from 3429 to 8000 per second from the ISP to the end user. The number of bits per symbol is limited by the channel's remaining frequency limitations and the signal power limits imposed by regulatory agencies (to limit cross talk in phone-wire trunks).

This opportunity has spawned diverse approaches. In early 1997, two similar but incompatible proprietary specifications exist: X2 and K56flex. Work is proceeding toward a first-stage international standard: V.pcm. This work will be completed early in 1998.

Addressing Cost of Ownership for Modems. The two largest cost-of-ownership issues for modems are installation problems and operations problems related to creating connections. Plug and Play minimizes installation problems when correctly implemented. However, the explosion in ISP usage has highlighted operations problems. According to public studies, 16.2 percent of ISP access calls fail to connect, and ISPs are commonly spending $6 per subscriber per month in technical support.

A smaller percentage of access calls fail after the connection is made. This failure rate is not acceptable. Three elements needed to change this are:

- Modem and PSTN diagnostics.
- Modems that can easily be upgraded.
- Deterministic modem identification so that upgraded modems still work.

In consultation with leading ISPs and modem manufacturers, Microsoft is developing a standard method for modems to report last-call statistics: the Unimodem Diagnostics command, or AT#UD, as described in the specification on http://www.microsoft.com/hwdev/specs/. This command will be used by Windows and ISP software. The reported last-call information is essential to illuminate problems in user modems, local phone loops, local offices, and ISP-side modems so that they can be diagnosed and fixed.

After a problem has been found, requiring users to replace their modems to fix that problem is too costly for both the user and the manufacturer. Some manufacturers already make their modems with upgradable memory, allowing painless feature or bug-fix upgrades for their customers. The makers of Windows-based modems have also offered this advantage. It is time to make easy user upgradability an industry-wide standard.

But even upgrades can pose hazards. For modems that do not support Plug and Play, the Windows Modem class installer reads a series of AT commands, implements a proprietary algorithm to generate a 32-bit ID, and uses that ID to match to the modem driver. Manufacturers might inadvertently change responses that Unimodem depends on for computing the unique Unimodem ID (ATI and other commands), leaving the user with a modem that is recognized as a “Standard Modem” instead of the actual modem name.

To address the detection problem, modem vendors are required to use bus-specific Plug and Play means to deliver the CompatibleID command, and they are encouraged to use standard methods to report accurate manufacturer and modem names. For information, see specifications for new Unimodem commands and related articles at http://www.microsoft.com/hwdev/hwdev/devdes/.

Migrating Functions to the Operating System with Windows Modems. A traditional modem has several functions implemented in hardware or firmware:

- Telephone network connection—connectors, transformers, relays, codec
- Digital signal processing—V.34, V.32bis, V.8bis, dual tone multifrequency (DTMF), voice processing, speakerphone echo cancellation, and so on
- Modem controller—AT command interpreter (for example, V.25ter)
- Protocol stacks—V.42 error control, V.42bis data compression

A Windows modem moves some of these functions into Windows drivers. A controllerless modem (also known as a host-based controller) is a modem that consists of a digital signal processor (DSP) without the usual microcontroller. The host CPU provides the AT command interpreter, modem control functions, V.42, and V.42bis implementation.

A software modem (also known as host-based signal-processing modem or pumpless modem) performs signal processing on the host Pentium processor or compatible microprocessor and implements the controller as described in the previous paragraph. The modem hardware consists only of a telephone-line interface and DAC and ADC circuitry such as a PCM codec, plus a little bit more. However, the hardware does not contain DSP or a microcontroller.

Advantages of software modems include the following:

- Cost savings
- Flexibility for upgrading to V.pcm, fixing bugs, and so on
- Data processing occurs in the CPU, where it fits
- Separate data and control paths to the hardware are available

Disadvantages of software modems include the following:

- CPU-based functions compete for resources with other uses, such as the operating system, applications, multimedia codecs, and so on
- Difficult to support multiple platforms, including Windows 3.1, Windows 95, Windows NT 3.*x*, and Windows NT 4.0, in addition to current and future versions of the Windows and Windows NT operating systems
- Difficult for users to install retail versions
- Difficult for automatic upgrade during operating system upgrade

Controllerless and software modems are built as custom VxD drivers for Windows and as custom system (.Sys) drivers for Windows NT; each platform requires a different driver. For PC 98, WDM Modem support will provide a common interface so that one driver runs on both Windows 98 and Windows NT 5.0.

Missing Features in Voice Modems. Voice modems are devices that allow a PC to communicate with a speaking and listening human at the other end of a telephone line. These modems digitize the incoming voice for recording; play back digitized audio; and detect various signals such as DTMF, call progress tones, and so on.

Voice capabilities started appearing in modems in the early 1990s, aided by the completion and publication of TIA IS-101 in 1993—the interim standard for Voice Data Communications Equipment (DCE). Now the trial period is over and the TIA-695 standard is complete. In mid-1997, at least 30 percent of modems provided for retail sales were voice capable, with a larger percentage in OEM sales. The following trends are important:

- Computer Telephony (CT). ISVs are figuring out what voice modems need.
- Voice-only devices emerging as a separate category (that is, CT telephones).
- Local handset or telset audio.
- Full-duplex audio for a number of applications.

The first applications for voice modems were focused on telephone answering machine (TAM) functions, such as Microsoft Phone. Some applications, such as Microsoft Operator, use voice modems for adaptive incoming-call selection. Now that the software community has had a chance to experiment, the interest in voice modems has expanded to include the following:

- Speakerphone capabilities, either through the PC or outside of it
- Low-end interactive voice response (IVR) equipment (for example, play voice while performing speech recognition and speaker recognition)
- IP telephony bridging
- Voice-only modems, some of which are integrated with a PC-connected telephone instrument (telset)

System Requirements for Modems

This section summarizes the PC 98 system requirements for modems.

1. Modem device is provided with PC system

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Required	*Required, if no network adapter*	*PCM modem required, upgradable to V.pcm*

Recommended: Internal modem, or use USB or PC Card as the external modem connection.

Modem Basic Features

This section defines basic PC 98 hardware feature requirements for modems.

2. Modem supports TIA-602 Hayes-compatible command set

Required

TIA-602 codifies the most common data modem. As part of the command set, the modem must have a distinct means for controlling the reporting of modem-to-modem protocols, so the modem can be forced to generate recognizable reports for modulation, error control, and data compression. Examples include the Wn command, the S95 register, the \Vn command, and the ITU V.25ter +MR, +ER, and +DR commands.

Recommended: ITU V.25ter, which is a superset of TIA-602 with significant and useful improvements. It includes three new components:

- A standard format for extending the AT command set, with standard means for the PC to test the range of supported values for each command. This enables adaptive modem installation.
- Standard extensions for modem ID, port control, modulation control and reporting, error control, and data compression control and reporting. This reduces or eliminates the need for data-modem INF files.
- Annex A (1996) provides standard commands that enable the PC to use V.25, V.8, and V.8bis call-control features for point-to-point data calls and voice/data/video calls.

The particular utility of the standard format is that it allows a future modem installer to adaptively install and use a modem, with minimal need for INF file minidrivers. This will be provided in versions of the MDK available in the second half of 1997.

3. Data modem supports 33.6 Kbps (V.34-1996) with V.42 and V.42bis protocol
Required
This is the minimum modem capability specified for PC 98.

4. Data modem supporting speeds faster than 33.6 Kbps can be upgraded to V.pcm
Required
Any modem that supports speeds faster than 33.6 Kbps must be capable of being upgraded to V.pcm. For Entertainment PC 98 systems, any modem supplied with the system must have this capability.

See the related “PCM modem supports ITU-T V.pcm” requirement later in this chapter.

5. Fax modem supports 14.4 Kbps (V.17) with Class 1 (TIA-578-A) command set
Required
The following are recommended:

- Class 1.0 (ITU T.31) with +FAR support, which allows the hardware to perform adaptive carrier detection
- Class 2.0 (ITU T.32 or TIA-592) for rack-mounted server modems
- Adaptive DATA/FAX call classification based on the Class 2.0 +FAA command or equivalent, particularly for rack-mounted server modems

Windows is bundled with fax-modem support. Windows NT 5.0 and future versions of Microsoft BackOffice® products will support Class 1.0 and Class 2.0 fax modems and use adaptive call-classification support. To benefit from this support, modem vendors should extend their modem INF files to support the new registry keys that are needed to support these features, as defined in the MDK available in the last quarter of 1997.

6. Data modem supports V. 80 for synchronous access
Recommended
This standard provides a control plane for the modem and synchronous access to the data path, the foundation for third-party voice/data/video software (for example, H.324, V.70). Modems that support V.80 should also include V.8bis signaling and V.25ter Annex A as described in the following item.

7. Modem supports adaptive connection, V.25, V.8, and V.8bis call control signaling with V.25ter Annex A modem commands

Recommended

V.25 defines basic call-type selection, with an answer tone, a fax-calling tone, and a data-calling tone. V.8 defines advanced call-type selection, with complex information exchanged between terminals; V.8 is used in V.34, V.pcm, and some digital simultaneous voice/data (DSVD) implementations. V.8bis is required for the following standard multimedia modes: V.61 ASVD, V.70 DSVD, and H.324 video telephony.

V.25ter Annex A enables the PC to participate in call control, allowing flexibility and a visual user interface as well as saving modem complexity.

8. Modem supports delayed and blacklisted number clearing

Recommended

The modem should clear its delayed and blacklisted number tables if the associated handset goes off hook.

During certain international PTT certification processes, modems must support the delayed and blacklisted numbers feature. That means that when the modem fails to connect to a specific number for a certain number of times, the dialed number is stored in an internal list. Any subsequent automated dialing operation to this number is then either delayed for a time (delayed) or might be forbidden until some form of manual intervention occurs (blacklisted). The international certification processes specify that manual intervention using an external device is required in order to clear these numbers.

Windows will provide error messages corresponding to delayed and blacklisted error reports in order to reduce customer confusion.

9. Modem supports TDD, meeting V.18-1996 with V.25ter AT commands

Recommended

People with deafness or reduced hearing can use Telephone Device for the Deaf (TDD), also known as Text Telephones, to communicate over phone lines. The U.S. Americans with Disabilities Act (ADA) requires all businesses of a certain size or larger to have Text Telephone services available and to be able to receive calls from people using Text Telephones.

In North America and Europe, the following types of Text Telephones are used:

- Baudot: 45 or 50 bps Frequency-Shift-Keyed (FSK) and 5-bit Baudot coding
- ASCII: 300 bps Bell 103 and 7-bit ASCII coding
- European Deaf Telephone (EDT): 110 bps half-duplex V.21 and 7-bit coding
- Minitel: V.23 modems and 7-bit coding
- Modems and 7-bit coding
- DTMF: 2-digit or 3-digit character coding

ITU recommendation V.18 codifies how all these devices work and how to adaptively connect to all of them. ITU recommendation V.25ter contains AT commands for control of V.18 features in a modem.

It is recommended to include Text Telephone capability for the type commonly used in the country of sale and use (for example, Baudot in the United States, Minitel in France, and so on).

10. PCM modem supports ITU-T V.pcm
Required
For PCM modems (faster than V.34-1996), ITU V.pcm is a requirement. Compliance testing for this requirement will begin within a reasonable time frame after ITU V.pcm is approved by ITU Study Group 16 (currently projected for January 1998). If V.pcm has not been approved by the time PC 98 compliance testing begins, the modem must be end-user upgradable by software means to the current version of V.pcm. It is critical not to strand end users with devices that must be replaced later.

11. Modem controller meets PC 98 requirements
Required
The following are PC 98 requirements for the modem controller:

- Unimodem Diagnostics command, AT#UD
- Software-upgradable modem controller (that is, upgradable ROM or Windows modem)
- AT command buffer of at least 60 characters
- Semicolon (;) character dial string modifier
- Connection reporting: DCE rate; Error Control, and Data Compression

The following are PC 98 recommendations for the modem controller:

- V.25ter +GMI and +GMM commands for modem identification. This is useful if the modem is installed with a CompatibleID.
- V.25ter +I, +M, +E, and +D commands. This allows automated generation of data modem INF file registry entries.

Voice Modem Requirements

Voice capabilities are not mandatory, but if present, the following requirements and recommendations apply.

There is a separate category of voice-only modem that can be integrated with a telset. These devices are not required to support data or fax, but the following requirements do apply.

12. Voice modem supports TIA-695 (AT+V)

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Required*

TIA IS-101-1994, the interim standard for Voice DCE, has been superseded by TIA-695. TIA-695 adds voice formats and speakerphone control commands. ITU-T plans an equivalent recommendation—V.voice—to be completed and assigned a number in January 1998. V.voice will add some small corrections to TIA-695, plus provisions for duplex voice. See also the "Voice modem supports full-duplex voice I/O" recommendation later in this section.

The following voice modem features are required:

- Voice recording and playback
- DTMF generation and detection during voice I/O
- Voice I/O support of 8-bit, 8-kHz PCM formats: unsigned linear, G.711
- Programmable gain control for all audio channels
- Speakerphone with automatic training (no user intervention)
- Voice I/O to the handset (for voice-only devices)

13. Voice modem support includes PC 98 recommendations
Recommended

The following voice modem features are recommended:

- Caller ID detection, reporting, and repeat (types 1 and 2)
- Sense local telephone line state (ready, busy, disconnected) without going off hook
- Extension telset answer and hang-up detection and reporting
- Programmable gain control for all audio channels
- Remote telset answer and hang-up detection and reporting

- Message waiting signal (stuttered dial tone) detection reporting
- Special Information Tone (SIT) detection and reporting
- Full-duplex voice I/O with echo cancellation, as described in the "Voice modem supports full-duplex voice I/O" requirement later in this section
- Distinctive ring detection and reporting
- Powered interface to the local telset to support voice I/O and DTMF I/O

14. Voice modem supports local telset interfaces

Recommended

For many voice applications, it is desirable for the PC and voice modem to be able to communicate with the person using the telset. This allows a consistent interface between local use and remote use, where remote means calling in from outside or using a cordless phone.

If the device is integrated with a telset, this is straightforward and a requirement; the handset and keypad are directly accessible within the device. If the device is not integrated with a telset, then there is a technical challenge. To gain access to the telset's handset and DTMF keypad, the voice modem must be able to cut the connection from the telset to the PSTN, and then must provide power and signal coupling to that telset.

The difficulty is that the switchable-isolation means (that is, the relay) must be able to withstand real or simulated lightning strikes of 1750 volts (U.S. Federal Communications Commission [FCC] Part 68) to 3750 volts (German FTZ) without catching fire. Therefore, this is a recommendation rather than a requirement for data/fax/voice modems.

15. Voice modem supports simultaneous voice/data integration capabilities

Recommended

A profusion of solutions have been offered for simultaneous multimedia phone calls, including propriety solutions, semi-public solutions (DSVD 1.2 and V.34Q), and standard solutions (ITU V.61, V.70-suite, and H.324-suite). Meanwhile, standard IP-based solutions are the wave of the future. Microsoft NetMeeting™, for example, uses H.323.

Therefore, PSTN voice/data integration is not required for desktops or servers. H.324 is emerging as a standard for consumer installations. To support H.324, modems should support ITU V.80, V.8bis, and V.25ter Annex A.

Note: VoiceView is not supported in Windows 98 or Windows NT 5.0.

16. Voice modem supports speakerphone

Recommended

Audio I/O for speakerphone can be implemented in any of the following ways:

- Built-in audio I/O (microphone and speaker). This support is appropriate for voice-only devices (that is, PC-connected phones).
- Jacks to external audio I/O (that is, microphone and speaker or handset jack). This is common for voice/data/fax modems with a large enough form factor to support the connectors.
- Full-duplex audio. See the "Voice modem supports full-duplex voice I/O" requirement later in this section. This support is appropriate for voice/data/fax modems and essential for PC Card modems, which might lack the connectors for external audio I/O.

17. Voice modem supports full-duplex voice I/O

Recommended

TIA-695 was written for TAM applications, with extensions for control of speakerphone functions. In the latter case, the speakerphone adaptive echo canceller is in the modem DSP, and the audio I/O is directed to separate microphone and speakers.

TIA-695 does not address simultaneous voice playback and recording. The applications that need this include speakerphones using audio I/O in the PC, playback while listening for voice responses, and PTSN-to-IP telephony bridging.

Microsoft is collaborating with other members of TIA TR-30 and ITU-T SG16 to define an addition to the draft recommendation V.voice (that is, TIA-695) to support full-duplex voice. This work could be completed by January 1998. Microsoft support for this feature will be based on that ITU-T recommendation.

Wireless and Cellular Modem Requirements

This section provides PC 98 requirements and recommendations for wireless and cellular modems.

18. Wireless support implemented for modems

Recommended

There is a variety of wireless modems and look-alike modems. These include the common types: North American analog cellular, global system for mobile communications (GSM) and other digital cellular systems, cellular digital packet data (CDPD), and so on. However, there are several other types, such as the Ricochet modem from Metricom.

Windows has registry keys that support analog cellular modems. Windows also supports data access in GSM and other wireless modem types. Participants in the Mobile Data Initiative are developing extensions for other services on digital cellular modems, as described in the following item.

For all wireless and cellular modems, the commands in TIA-678 are recommended. The +WS-46 command, which selects the wide area network (WAN), is required.

19. Digital cellular phone support is implemented for modems
Recommended
Digital cellular support is not a requirement, but if implemented, the following appropriate digital cellular control standards must be supported:

- TIA-678 +WS-46 selector command
- Class 2.0 facsimile services, per appropriate standard

Unlike wireline data modems, these devices are not required to support V.34 signaling; it is not available. 9600 bps capability is required; higher speeds are recommended where available. Class 1.0 fax support is available on some of these devices, but it is not required; the error rates with transparent modem faxes are often very high.

Cellular telephone systems are widely deployed in the industrialized world and are now being deployed internationally. In North America, analog cellular systems (TIA-553) are currently predominant, although two types of digital cellular systems can also be deployed: code division multiplexed access (CDMA, TIA IS-95) and time division multiplexed access (TDMA, IS-136).

In Europe and the rest of the world, the GSM digital cellular system is widely deployed. In Europe, the infrastructure for data, fax, and short messaging is now in place.

For all three digital cellular systems, the system design has been extended to offer data, fax, voice, and short messaging service (SMS) to mobile users. In all cases, a modem pool is added to the ground stations, where connection is made to the PSTN. Access to the logical serial ports of these modems is made using the digital error-controlled radio link to the equipped mobile phone and is exposed on a serial port or associated PC Card.

Digital cellular communications equipment should default to using error correction on the radio link. For example, for GSM 7.07, the modem should initialize to +CBST=,,1 (which selects a “nontransparent” air interface).

The AT command sets for these digital cellular phone systems are contained in the following standards.

Standard	Command set
GSM 7.07	GSM system: data, fax, voice
GSM 7.05	GSM SMS
TIA IS-99	North American CDMA: data and fax
TIA IS-135	North American TDMA: data and fax

The TIA-678 +WS46 command has codes to indicate which system the modem is capable of. For example, the following values, quoted from Table 4 of the standard, are useful.

Value	System
1	Public telephone network (that is, a normal wireline modem)
4	Cellular Digital Packet Data (CDPD)
7	TIA-553 analog cellular system
10	Metricom Ricochet network
12	GSM digital cellular system
13	TIA IS-95 CDMA digital cellular
14	TIA IS-136 TDMA digital cellular ("PCS")

ISDN Modem Requirements

There are two classes of ISDN adapters: (1) parallel bus devices, supported by NDIS WAN drivers, and (2) serial port devices, supported by Unimodem with INFs. This section addresses serial ISDN modems.

For a general discussion of ISDN and a list of requirements related to parallel bus devices, see the "Network Communications" chapter in Part 4 of this guide.

ISDN modems share the following features:

- ISDN Basic Rate interface (2B+D)
- Serial AT command language, with proprietary ISDN extensions

ISDN modems also share the following differences from wireline PSTN modems:

- User (or device) must configure for switch type and service profile ID (SPID)
- Data only, in increments of one or two 64,000 bps B channels
- Fax not available
- V.42 and V.42bis usually not available

20. ISDN modem supports required command set

Required

An ISDN modem must support basic AT commands (TIA-602, which is a subset of ITU V.25ter). Also, commands to set the switch type and SPID for user selection or if auto-detection fails must be included. This can be implemented in the device or in the communications driver.

21. ISDN modem supports auto-SPID detection algorithms and standard SPID format

Required

An ISDN modem must include commands or means to support software-based automatic switch type and SPID detection using the algorithms as defined by the National ISDN User's Forum (NIUF) in *1997 Version of National ISDN Basic Rate Interface Terminal Equipment Generic Guidelines.* This eliminates the need for the end user to enter the SPIDs.

An ISDN modem must include commands or means to support software-based automatic switch type and SPID detection using the algorithms defined by NIUF. This eliminates the need for the end user to enter the SPIDs and enhances the Plug and Play experience for users.

This requirement applies only in the United States.

22. ISDN modem supports CHAP in firmware if B channels are not exposed

Required

Recommended: Support MS-CHAP.

If the ISDN modem implements support for multilink point-to-point protocol (PPP), it must also support Challenge Handshake Authentication Protocol (CHAP). Supporting multilink PPP on an ISDN modem requires the devices to authenticate the second call themselves.

The remote access server uses CHAP to negotiate the most secure form of encrypted authentication supported by both server and client.

23. ISDN modem exposes both B channels

Recommended

ISDN modems should expose both B channels so that they can support the multilink PPP stack.

External ISDN modems should be on port fast enough to expose the full bandwidth of both B channels, that is, USB. Providing two separate COM-port cables is not an acceptable solution.

24. ISDN modem supports multilink PPP

Recommended

Multilink PPP as defined in RFC 1717 combines several ISDN B channels to increase the bandwidth of PPP links.

Windows and Windows NT operating systems include support for multilink PPP. When using ISDN modems that appear as modems to the operating system, multilink PPP must be implemented in the device. This is because Windows cannot see both B channels of the ISDN connection unless each B channel is exposed as a COM port.

See also the “ISDN modem supports CHAP in firmware if B channels are not exposed” requirement earlier in this section.

This recommendation is for ISDN modems only. Internal ISDN devices with NDIS WAN miniport drivers benefit from the built-in multilink support provided by the remote access services of the operating system, and therefore do not need to provide multilink PPP support.

25. ISDN modem supports asynchronous-to-synchronous conversion

Required

These types of ISDN devices are treated as modems, not as internal ISDN devices supported using NDIS WAN miniports. In the external case, the primary implication is that the operating system will send byte-level PPP (also known as asynchronous PPP). In the NDIS WAN case, the implication is that the operating system will send bit-level PPP (also known as synchronous PPP).

Because ISDN is a synchronous service and an ISDN modem connects to an asynchronous port on the PC, the device must provide some means of converting asynchronous data to synchronous data.

26. ISDN modem uses high-speed port

Recommended

Because of speed limitations inherent in a PC’s COM ports, the connection for ISDN modems should be high-speed, such as USB.

27. ISDN driver supports switch detection

Recommended

The driver can attempt to determine the switch type based on the directory number, or it can use other proprietary solutions to determine the switch type. This enhances the Plug and Play experience for users.

28. ISDN driver supports unattended installation, with limitations
Required
ISDN devices must be capable of being installed without user intervention. The exception is specific ISDN parameters, which must be acquired from the equipment being connected to. Dependent parameters include SPIDs and switch-type IDs.

PC 98 Design for Modems

This section summarizes PC 98 requirements related to the design initiatives in Part 1 of this guide.

Plug and Play and Bus Design for Modems

The items in this section are requirements for Plug and Play capabilities.

29. Each device has a unique Plug and Play device ID
Required
For a system-board device, there must be a device-specific ID.

Each bus-specific device must provide device IDs in the manner required for the bus it uses as defined in Part 3 of this guide. For example, PCI devices must comply with PCI 2.1 requirements and also must provide a Subsystem ID and Subsystem Vendor ID as defined in the "PCI" chapter in Part 3 of this guide.

Note: The device must implement either a bus Plug and Play ID or a COM-port Plug and Play ID, but not both.

30. Each device has a compatible Plug and Play device ID
Required
The various bus-specific Plug and Play specifications provide the means for reporting a CompatibleID as well as a device unique ID.

At least one CompatibleID is required for PC 98. Its primary use is for back up in case the driver or INF file associated with the unique ID is not available (for example, if the customer lost the disk). The goal is for the modem to retain essential data functionality.

The most useful CompatibleIDs would point either to an earlier version of the same product (whose INF file is included in shipping versions of Windows) or point to a reference INF file (that is, one provided by the modem chip-set manufacturer).

Note: If the CompatibleID is used, it is recommended to provide an accurate, displayable manufacturer and modem name using V.25ter standard ID commands.

31. Automatic resource assignment and dynamic disable capabilities are supported

Required

The system must be capable of automatically assigning, disabling, and relocating the resources used by this device when necessary, using the method required for the related bus class. When the end user changes this device or adds it to the system, setting resource assignments must not require changing jumpers or switches on either the adapter or the system board. In the event of an irreconcilable conflict with other devices on the system, the system must be able to disable the device to prevent the system from stalling.

32. PCI modem meets PCI 2.1 requirements

Required

This device must comply with PCI 2.1 requirements if PCI is used as the bus connection for the modem. This ensures that all Plug and Play requirements are met and that Windows and Windows NT drivers support this device.

33. USB modem meets USB specifications

Required

As required for all PC 98 devices, a modem must meet the specific requirements for the bus it uses, including any device class specifications. For example, a modem that uses USB must comply with all related USB specifications, including:

- *USB Specification, Version 1.0* or higher (also known as the USB core specification)
- *Universal Serial Bus Device Class Definition for Communication Devices, Version 1.0* or higher

  The “Standard Serial Interface Circuit Emulation” appendix in the *USB Device Class Definition for Communication Devices* specifically addresses serial port compatibility.

34. Device Bay modem meets PC 98 requirements

Required

A modem designed as a Device Bay peripheral must interface with either USB, IEEE 1394, or both buses, and must support relevant USB device class specifications. All Device Bay peripherals must meet the requirements defined in *Device Bay Interface Specification, Version 1.0* or higher.

Power Management for Modems

This section summarizes the modem power management requirements. See also the specific power management requirements for each bus defined in Part 3 of this guide.

35. Device complies with device class power management reference specification
Required

The *Communications Device Class Power Management Reference Specification, Version 1.0* or higher, provides definitions of the OnNow device power states (D0–D3) for these devices, including modems. The specification also covers the device functionality expected in each power state and the possible wake-up event definitions for the class.

Power states D0 and D3 are required for modems on power-managed buses, including PCI, CardBus, and USB.

36. Device supports wake-up events
Required

For PC 98, a required modem feature is the ability to cause a wake-up event on an incoming ring as defined in *Communications Device Class Power Management Reference Specification.* Notice that this applies for modems on power-managed buses, including PCI, CardBus, and USB.

The D2 power state is defined specifically for this purpose in the power management reference specification. The ability for a modem to cause a wake-up event from the D3 power state might also be possible and is recommend to realize better system power savings. To comply with this requirement, a modem must be able to cause a wake-up event from either the D2 state, the D3 state, or both states.

Because caller-ID reporting would be missed by PCs while in a sleep state, the ability for a modem to retain and repeat the last caller-ID reporting on demand is strongly recommended. The mechanism for doing this is described in *Communications Device Class Power Management Reference Specification* and in the V.voice and TIA-695 voice modem specifications.

Device Drivers and Installation for Modems

This section summarizes device driver requirements for modems. The items in this section are requirements for all PC 98 systems.

37. Device drivers and installation meet PC 98 requirements

Required

The manufacturer does not need to supply a driver if a PC 98-compliant driver provided with the operating system can be used. If the manufacturer supplies a driver, the requirements for the device drivers and installation are defined in the "Basic PC 98" chapter in Part 2 of this guide. The basic requirements include driver support for unattended installation and Help file support if special driver parameters are used.

For information about support for controllerless and software modems under WDM, see the Windows NT 5.0 DDK. See also the related articles at http://www.microsoft.com/hwdev/pcfuture/. For information about WDM-based modem driver support under Windows NT, see the Windows NT 5.0 DDK.

For guidelines about implementing driver and installation support for modems under the Windows operating system, see the Windows MDK.

38. Driver supports Unimodem

Required

The device driver must include Unimodem support. Typically, this requires a modem INF file, developed and verified using the MDK and pretested by the modem manufacturer.

39. Applications provided with device meet Win32 requirements

Required

Any Windows-based applications provided with the device, such as fax utilities, must meet requirements for software compatibility as defined in the Win32 SDK.

As an API, the Windows Telephony API (TAPI) is the cornerstone of telephony for Windows and Windows NT. Telephony applications and service providers provided with PC 98 systems must be implemented using TAPI 2.0.

Among other enhancements, applications can request, negotiate, and renegotiate QOS parameters with the network and receive indication of QOS on inbound calls and when QOS is changed by the network. For a summary of the TAPI 2.0 architecture and a description of how to write a TAPI service provider, see http://www.microsoft.com/ntserver/communications/. For implementation information, see the Win32 SDK.

Modem References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

ANSI, EIA, TIA, and other standards
Global Engineering Documents
Phone: (800) 854-7179 (US)
(613) 237-4250 (Canada)
(303) 792-2181 (Outside North America)
Fax: (303) 397-2740
ftp://ftp.symbios.com/pub/standards/io/

Bellcore Technical References
Bellcore (Bell Communications Research)
Phone: (800) 521-2673 (North America)
(908) 699-5800 (Outside North America)
http://www.bellcore.com

Communications Device Class Power Management Reference Specification, Version 1.0
http://www.microsoft.com/hwdev/onnow.htm

Device Bay Interface Specification, Version 1.0
http://www.device-bay.org

European Telecommunications Standards Institute (ETSI) or Global System for Mobile (GSM) standards
Phone: +33-92 94 42 00
FAX: +33-93 65 47 16
E-mail: secretariat@etsi.fr

ITU communications standards
ITU Sales
Phone: (41) (22) 730-6141
Fax: (41) (22) 730-5194
E-mail: sales@itu.ch
http://www.itu.ch

PCI Local Bus Specification, Revision 2.1 (PCI 2.1)
http://www.pcisig.com

Plug and Play specifications
http://www.microsoft.com/hwdev/specs/

Telephony API (TAPI) overview and white papers
http://www.microsoft.com/ntserver/communications/
http://www.microsoft.com/win32dev/netwrk/tapiwp.htm

Unimodem Diagnostics Command Reference Specification
Unimodem ID Command Reference Specification
http://www.microsoft.com/hwdev/specs/

USB specifications
http://www.usb.org

WDM device driver support white papers
http://www.microsoft.com/hwdev/pcfuture/

Windows MDK
ftp://ftp.microsoft.com/developr/drg/modem/modemdev.exe

Windows and Windows NT DDK, including information about NDIS and Win32 SDK
MSDN Professional membership

Checklist for Modems

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

Consumer PC 98	Office PC 98	Entertainment PC 98
1. Modem device is provided with PC system		
Required	*Required, if no network adapter*	*PCM modem required, upgradable to V.pcm*
2. Modem supports TIA-602 Hayes-compatible command set		
Required		
3. Data modem supports 33.6 Kbps (V.34-1996) with V.42 and V.42bis protocol		
Required		
4. Data modem supporting speeds faster than 33.6 Kbps can be upgraded to V.pcm		
Required		
5. Fax modem supports 14.4 Kbps (V.17) with Class 1 (TIA-578-A) command set		
Required		
6. Data modem supports V. 80 for synchronous access		
Recommended		
7. Modem supports adaptive connection, V.25, V.8, and V.8bis call control signaling with V.25ter Annex A modem commands		
Recommended		
8. Modem supports delayed and blacklisted number clearing		
Recommended		
9. Modem supports TDD, meeting V.18-1996 with V.25ter AT commands		
Recommended		
10. PCM modem supports ITU-T V.pcm		
Required		
11. Modem controller meets PC 98 requirements		
Required		
12. Voice modem supports TIA-695 (AT+V)		
Recommended	*Recommended*	*Required*

13. Voice modem support includes PC 98 recommendations
Recommended

14. Voice modem supports local telset interfaces
Recommended

15. Voice modem supports simultaneous voice/data integration capabilities
Recommended

16. Voice modem supports speakerphone
Recommended

17. Voice modem supports full-duplex voice I/O
Recommended

18. Wireless support implemented for modems
Recommended

19. Digital cellular phone support is implemented for modems
Recommended

20. ISDN modem supports required command set
Required

21. ISDN modem supports auto-SPID detection algorithms and standard SPID format
Required

22. ISDN modem supports CHAP in firmware if B channels are not exposed
Required

23. ISDN modem exposes both B channels
Recommended

24. ISDN modem supports multilink PPP
Recommended

25. ISDN modem supports asynchronous-to-synchronous conversion
Required

26. ISDN modem uses high-speed port
Recommended

27. ISDN driver supports switch detection
Recommended

28. ISDN driver supports unattended installation, with limitations
Required

29. Each device has a unique Plug and Play device ID
Required

30. Each device has a compatible Plug and Play device ID
Required

31. Automatic resource assignment and dynamic disable capabilities are supported
Required

32. PCI modem meets PCI 2.1 requirements
Required

33. USB modem meets USB specifications
Required

34. Device Bay modem meets PC 98 requirements
Required

35. Device complies with device class power management reference specification
Required

36. Device supports wake-up events
Required

37. Device drivers and installation meet PC 98 requirements
Required

38. Driver supports Unimodem
Required

39. Applications provided with device meet Win32 requirements
Required

CHAPTER 20

Network Communications

20

This chapter presents PC 98 requirements and recommendations for network adapters and related technologies.

Tips for Selecting High-Performance Network Adapters. For PC manufacturers who want to select high-performance components, the following are design features to look for in network adapters:

- Adapter supports bus mastering, especially for use with Pentium Pro processors
- PCI adapter properly supports higher-level PCI commands for intelligent data transfer
- Drivers are tuned for 32-bit performance

Contents

Introduction to NDIS 5.0

The Network Driver Interface Specification (NDIS) 5.0 represents a number of extensions to the interface described in NDIS 3.0 and 4.0. The basic requirements, services, terminology, and architecture of the earlier versions also apply to NDIS 5.0. The new NDIS architecture will be included in Windows 98 and Windows NT 5.0 operating systems.

NDIS 5.0 consists of all functionality defined in NDIS 4.0, plus the following extensions:

- NDIS power management, required for network power management and network wake up.
- Plug and Play, which was previously supported only under Windows 98 but is now also applicable to Windows NT 5.0 network drivers.
- Windows hardware instrumentation support for structured, cross-platform management of NDIS miniports and their associated adapters.
- Simplified network INF format across Windows operating systems, based on the Windows 95 INF format.

  The goal for the new-style INF file format is to enable a single INF file to work on both Windows and Windows NT.
- Deserialized miniport for improved performance on Windows NT multi-processor systems.
- Task-offload mechanisms for tasks such as TCP/IP checksum, PPP compression/encryption, and Fast Packet Forwarding.
- Broadcast media extension, required for broadcast components.
- Connection-oriented NDIS, required for native access to connection-oriented media such as ATM and asymmetric digital subscriber line (ADSL), and WDM support for streaming over connection-oriented media.
- Support for QOS when supported by the media.
- Intermediate driver support, which is required for broadcast components, virtual LANs, LAN emulation over new media (ATM, satellite or broadcast television, and so on), packet scheduling for QOS, and NDIS support over WDM-supported buses such as IEEE 1394.

NDIS Extensions for ATM and QOS. Previously, NDIS primarily supported network interface card driver development and deployment of connectionless network media such as Ethernet, Token Ring, ArcNet, and Fiber Distributed Data Interface (FDDI). NDIS 5.0 extends this interface to provide efficient support for connection-oriented media such as ATM and ISDN, with support for QOS and with isochronous data transfer for media that supports QOS. The new architecture also enables support for streaming of multimedia data such as audio and video over the NDIS media.

Information about the miniport driver model is included in the Windows NT 5.0 DDK.

Note: NDIS 5.0 features are accessible only by using the NDIS miniport driver model and are not supported for full MAC drivers.

System Requirements for Network Communications

This section summarizes the basic hardware design features for network communications devices and the specific features for PC 98.

1. PC system includes network adapter

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Required, if no modem*	*Recommended*

It is recognized that OEMs supply PC systems to corporations for networking purposes in situations where the customer will insert network adapters at the end-user site. If the device is present in the system, it must meet the minimum requirements for network adapters defined in this chapter. Office PC 98 systems submitted for testing must include either a network adapter or a modem.

2. PC system includes internal or external ISDN device

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Recommended*

If an ISDN device is present in the system, it must meet the minimum requirements for ISDN devices defined in this chapter. For information about serial ISDN devices, see the "Modems" chapter in Part 4 of this guide.

3. PC system includes cable modem

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Optional	*Optional*	*Optional*

Cable modems are not required features for any PC 98 system. Recommendations are provided in this chapter for informational purposes only.

4. PC system includes ATM adapter

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Optional	*Optional*	*Optional*

ATM adapters are not required features for any PC 98 system. If the device is present in the PC system, it must meet the minimum requirements for ATM adapters defined in this chapter.

5. PC system includes ADSL adapter

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Optional	*Optional*	*Optional*

ADSL devices are not required features for any PC 98 system. If the device is present in the PC system, it must meet the minimum requirements for ADSL adapters defined in this chapter.

6. PC system includes satellite or broadcast receiver with NDIS driver

Consumer PC 98	*Office PC 98*	*Entertainment PC 98*
Recommended	*Recommended*	*Recommended*

For information about the PC 98 requirements for supporting a broadcast receiver, which requires NDIS 5.0 support, see the "Video and Broadcast Components" chapter in Part 4 of this guide.

Network Adapter Requirements

This section defines basic PC 98 hardware feature requirements for network adapters. Many of these requirements also apply for other network communications devices described in this chapter.

7. Adapter uses NDIS 5.0 miniport driver
Required

For PC 98, the network adapter driver must support NDIS 5.0 in order to take advantage of new operating system capabilities and must follow the NDIS miniport driver model. A full MAC implementation is not compliant with PC 98 requirements.

If the device is a switched WAN card (for example, ISDN, X.25, or S56), it must have an NDIS WAN miniport driver that supports all TAPI functions as defined for NDIS 5.0 in the Windows NT 5.0 DDK. It must support Windows remote access or other WAN services over WAN media.

8. Full-duplex adapter automatically detects and switches to duplex mode
Required

If the network adapter supports full duplex and if the network switch the adapter is connected to supports full duplex and standard ways of detecting the duplex mode, then the network adapter must be capable of automatically detecting the duplex mode and must use that mode. The goal is to automatically configure this setting without end-user intervention.

9. Adapter automatically senses presence of functional network

Required

The network adapter must be capable of dynamically determining whether it is functionally connected to a link partner such as a hub, switch, or router. If the adapter is on an expansion card not used as a boot device, then the device drivers can determine the presence of the functional link. If it is not functionally connected to a link partner, the miniport driver must provide appropriate NDIS status indication, using support for cable sense in NDIS 5.0.

No time duration is specified for the required detection or status indication.

For information about NDIS status codes and indication mechanisms, see the Microsoft Windows NT 5.0 DDK.

10. Adapter automatically senses transceiver type

Required

Network adapters that support multiple transceivers must be capable of automatically detecting which transceiver type is connected to the network. The network adapter then must automatically use that transceiver. In all cases, the user must not be required to set jumpers or manually enter information to inform the operating system of the transceiver type.

11. Adapter supports quadword buffer alignment for receive and byte buffer alignment for send

Required

Recommended: Byte buffer alignment for all buffers, which imposes fewer limitations on overlying software components.

Buffer alignment refers to the allowed offset addresses (boundaries) where packets can begin. Memory allocated for receive buffers must be quadword-aligned or better (byte-aligned or word-aligned). The send buffer must be capable of handling byte-aligned buffers.

The network adapter must impose minimal buffer-alignment restrictions.

12. Adapter communicates with driver across any bridge

Required

If the adapter uses a bridge, all communications must be free of errors across any bridge, such as a PCI bridge adapter.

13. Adapter supports filtering for 32 multicast addresses, at minimum

Required

Recommended: 128 addresses.

This capability is needed to support new push technology applications such as Microsoft NetShow, Active Desktop, and Internet Explorer 4.0. PC 98 requires a minimum capability for filtering 32 multicast addresses (also known as channels). This number is expected to increase incrementally in coming years. Future requirements will specify filtering for a minimum of 128 addresses.

14. Adapter is compatible with remote new system setup capabilities if used as boot device

Required

For a PC 98 system that uses a network adapter to support installing the operating system, the network adapter must be compatible with remote new system setup capabilities as defined in the open industry-standard Dynamic Host Configuration Protocol (DHCP). The DHCP provides for dynamic configuration of PCs on TCP/IP networks, as specified in Internet Engineering Task Force (IETF) RFCs 1533, 1534, 1541, and 1542. Trivial File Transfer Protocol (TFTP), Revision 2, supports boot-image download, as specified in IETF RFC 1530.

For Net PC systems and Office PC 98, a network adapter and system BIOS support are required to use the adapter as a boot device, as defined in the "BIOS meets PC 98 requirements for boot support" requirement in the "Basic PC 98" chapter in Part 2 of this guide.

The complete mechanism for remote new system setup is defined in the *Network PC System Design Guidelines,* provided as Appendix E in the References part of this guide.

15. Device Bay network adapter meets PC 98 requirements

Required

Any networking communications device designed as a Device Bay peripheral must interface with either USB, IEEE 1394, or both, and must support relevant USB device class specifications. All Device Bay peripherals must meet the requirements defined in *Device Bay Interface Specification, Version 1.0* or higher.

ISDN Requirements

This section summarizes the design features for ISDN devices.

In this section, “internal ISDN device” refers to the ISDN terminal adapter, which exposes raw access to its B channels using NDIS miniports. NDIS miniports could also be attached to the PC using WDM-supported bus classes such as USB (thereby physically being an external device).

In this section, “ISDN modem” refers to an internal or external ISDN device that exposes itself as a modem controlled by the AT command set. Certain ISDN devices might be attached to the PC internally or externally, exposing serial ports or modems to the system. To Windows operating systems, these devices look like modems, and the operating system can use these devices as modems, provided that the hardware manufacturer has done the work to ensure that these devices look and act like modems. This work includes the following:

- Interpretation of the standard modem AT command set. This can be done either in the ISDN device itself or in a serial port driver. For more information, please refer to the TIA-602 specification, a subset of ITU V.25ter.
- A modem INF file for installing the device and for telling Unimodem which commands to use to control the ISDN device.

This section defines general requirements for ISDN and specific requirements for ISDN terminal adapters. For more information about the requirements for ISDN modems, see the “Modems” chapter in Part 4 of this guide.

For PC 98, ISDN is recommended but not required for high-speed connections. If implemented in a PC 98 system, ISDN must meet the requirements defined in this chapter. For Plug and Play, power management, and driver support requirements, see the “PC 98 Design for Network Communications” section later in this chapter.

16. Internal ISDN device meets PC 98 network adapter requirements
Required
For PC 98, the driver must support NDIS 5.0. Only NDIS 5.0 miniport drivers and INF files are allowed for complete user-friendly installation and operation of the ISDN adapter.

The following requirements must be met, as defined in the “System Requirements for Network Communications” section earlier in this chapter:

- Support NDIS 5.0 using a miniport driver
- Automatically sense whether a cable is connected
- Support quadword buffer alignment or better
- Communicate with a driver across any bridge

17. Internal ISDN device supports synchronous HDLC framing

Required

High-level data link control (HDLC) framing is a standard for sending synchronous data. Other framing methods are allowed if the miniport driver provides simple HDLC framed synchronous PPP packets to NDISWAN using NDIS.

18. Internal ISDN device uses NDIS WAN miniport driver

Required

The device must have an NDIS WAN miniport driver to support Windows remote access over ISDN. For information about NDIS status codes and indication mechanisms, see the Windows NT 5.0 DDK.

19. Internal ISDN device includes connection for analog phone

Recommended

Note: This recommendation applies only in the United States.

Adding an analog (POTS) port to the ISDN device delivers convenience to the small-office/home-office (SOHO) market, allowing customers to use one ISDN line to meet all their telecommuting needs at minimal cost. Many customers in this market don't want a separate analog phone line for their fax machines, modems, or phone when ISDN can do this with a device that has POTS support.

20. ISDN device supports auto-SPID detection algorithms and standard SPID format

Required

Note: This requirement applies only in the United States.

ISDN drivers must support auto-SPID detection algorithms as defined by NIUF in *1997 Version of National ISDN Basic Rate Interface Terminal Equipment Generic Guidelines*. This capability eliminates the need for the end user to enter the SPIDs.

ISDN drivers also must support the standard SPID format, which is useful when auto-SPID is not available.

21. ISDN driver supports switch detection

Recommended

The driver can attempt to determine the switch type based on the directory number, or it can use other proprietary solutions to determine the switch type. This enhances the Plug and Play experience for users.

22. ISDN driver supports unattended installation, with limitations

Required

ISDN devices must be capable of being installed without user intervention. The exception is specific ISDN parameters, which must be acquired from the equipment being connected to. Dependent parameters include SPIDs and switch-type IDs.

Notice that it is a PC 98 requirement for the device to support auto-SPID detection algorithms. If these algorithms are also supported on the equipment being connected to, only the switch type should have to be entered during installation of ISDN hardware if it cannot be detected automatically.

23. ISDN device includes built-in NT-1

Recommended

Note: This recommendation applies only in the United States.

NT-1 (network terminator) splits the duplexed transmit and receive signals from the ISDN line into separate transmit and receive components. An ISDN device with a built-in NT-1 can connect directly to the ISDN line. However, doing so prevents other devices from being attached to the ISDN line (only one NT-1 can be connected to an ISDN line).

Cable Modem Recommendations

Cable modems are neither required nor recommended for PC 98. The recommendations provided in this section are for informational purposes only.

Cable modems connected to a PC are one component in a system that cable-television operators use to deliver high-speed cable data services (HSCDS) to customers. HSCDS provides two-way services: Data flows downstream from the cable operator’s head end and upstream from the customer’s PC.

Ideally, two-way HSCDS is delivered over 450-MHz and 550-MHz hybrid fiber-coax (HFC) cable-television distribution networks. Most current cable-television distribution networks in the United States are 300-MHz, 350-MHz, and 400-MHz coax branch and tree networks.

Currently, the typical cable modems used on customer premises are external to the PC, with their own power supply. These cable modems are essentially IP routers. The cable modem’s upstream connection is either a standard F-coax interface or an ATM interface. The cable modem connection to the PC uses a 10BaseT interface (RJ45) to connect a twisted pair to a standard Ethernet network adapter. For the next several years, it is expected that external cable modems will be included as part of the HSCDS service by the cable operator, much like set-top boxes. In light of that, along with the absence of standards for cable modems, few PC vendors are expected to bundle cable-modem hardware with their systems.

PC platform vendors can build PCs that are capable of using external cable modems by including a standard Ethernet network adapter in their PC systems. The PC platform should also have pre-installed Microsoft TCP/IP software.

HSCDS upstream data flow can be implemented on the predominant 300-MHz, 350-MHz, and 400-MHz coax branch and tree cable-television distribution networks by using telephone circuits for the upstream channel. This is a design alternative used today to accomplish HSCDS field trials with existing cable-television networks until more HFC networks are built.

There are a great number of design issues for cable operators as they upgrade their cable-television distribution networks to HFC and begin to deliver HSCDS. Even operators who have existing HFC networks must add equipment to that network in order to begin offering HSCDS. The speed at which cable-television operators resolve these issues will determine the requirements for cable modems in homes and businesses in the next two years. Although HSCDS is only in trials today, a majority of the cable operators in the United States are working on business plans for delivering HSCDS within the next two years.

PC platform makers must track the rate at which cable-television operators are resolving these issues and must track the best methods. The key cable-modem design issues for PC platform makers are cost and standards.

Some cable-modem vendors plan to implement the new IEEE 802.14 protocol when it becomes available. It should be well-suited to the distances, data rates, and physical-plant scenarios for HSCDS on cable-television distribution networks.

If implemented under Windows or Windows NT, a cable modem should support PC 98 network adapter requirements, including Plug and Play, power management, and driver support.

The following recommendations should be considered for cable modems implemented in PC 98 systems:

- **Provide platform support for external cable modems.** Plan to support external cable modems by supplying a low-cost Ethernet network adapter as part of a PC until a low-cost internal cable modem becomes available.
- **Provide an Ethernet card solution for cable modems.** PC platform makers providing a standard Ethernet card to connect to an external cable modem can contribute today to a low-cost solution at the user’s end for HSCDS service.
- **Participate in developing standards for cable modems.** Microsoft encourages cable-modem vendors to contact Microsoft about the devices they are designing. Microsoft wants to work with the cable-television industry in establishing standards for the use of cable modems in HSCDS delivery networks.

ATM Adapter Requirements

The NDIS 5.0 extensions provide kernel-mode NDIS 5.0 client drivers with direct access to connection-oriented media such as ATM. The new architecture for Windows and Windows NT extends native ATM support to Windows Sockets 2.0 (WinSock) and DirectShow-based applications by providing system-level components that map the applicable WinSock and DirectShow APIs to NDIS 5.0, extending direct ATM access to user-mode applications.

If ATM is included in a PC 98 system or is specifically designed for Windows or Windows NT, it must meet the requirements outlined in this chapter. For basic requirements for Plug and Play, power management, and driver support, see the "PC 98 Design for Network Communications" section later in this chapter. For more details about the following requirements, please refer to Section 3, "ATM Layer Specification," in *ATM User-Network Interface Specification, Version 3.1*.

24. ATM adapter meets PC 98 network adapter requirements

Required

The following requirements must be met as defined in the "System Requirements for Network Communications" section earlier in this chapter:

- Support NDIS 5.0 using a miniport driver
- Automatically sense whether a network cable is connected
- Support quadword buffer alignment or better
- Communicate with a driver across any bridge

25. ATM adapter supports a minimum number of active connections

Required

The VPI (Virtual Path Identifier) and VCI (Virtual Channel Identifier) ranges supported by the adapter affect the maximum number of simultaneous connections supported on a system.

This affects the applicability of the adapter to ATM applications such as LAN emulation, where at least one dedicated VC is created between each pair of communicating ATM hosts.

System type	Simultaneous connections
Client	64 or more
Server	2048 or more

A sample driver is provided in the Windows NT DDK to guide developers in properly supporting resources to meet this requirement.

26. ATM adapter supports all service types defined by the ATM Forum

Recommended

The ATM adapter should support the constant bit rate (CBR), variable bit rate (VBR), available bit rate (ABR), and unspecified bit rate (UBR) service types as defined by the ATM Forum.

27. ATM adapter supports a minimum number of simultaneously active rt-VBR/nrt-VBR/CBR connections

Required

Support for at least two simultaneously active rt-VBR/nrt-VBR/CBR connections is required for basic applications, such as LAN Emulation (where “rt” stands for real time and “nrt” stands for non-real time). The two simultaneous active connections can be any combination of two from the three service types: rt-VBR, nrt-VBR, and CBR.

Support for additional CBR/rt-VBR connections is required for ATM adapters that support multimedia or other traffic that demands QOS. These are listed in the following table.

System type	Simultaneous active rt-VBR/nrt-VBR/CBR connections
Client	6
Server	500

28. ATM adapter supports traffic shaping

Required

The ATM adapter must support and enforce all the traffic-shaping rules specified for each service type it supports, including CBR, VBR, ABR, and UBR. For example, this includes enforcement of peak cell rate on a UBR virtual circuit.

If the ATM adapter is connected to residential broadband networks—either directly or using external devices such as an ADSL modem or cable modem—there are additional traffic shaping requirements that restrict the total transmission rate of all active virtual circuits not to exceed the upstream bandwidth of the residential broadband network.

To support these types of connections with possibly limited throughput or asymmetric speeds, the adapter must enforce traffic shaping of all (aggregated) ATM traffic on the line, based on maximum line rate and peak cell rate (PCR) for outgoing traffic. This rate normally will be read from the adapter and reported to NDIS. However, a new requirement is that NDIS must also be able to write a lower rate back to the adapter, which shall then shape the aggregate of all ATM traffic to this indicated PCR.

29. ATM adapter supports external clocking
Required

Recommended: ATM adapter supports both internal and external (default) clocking.

Usually adapters can derive their transmit clocks from the switch's Sonet frames (external clocking). Internal clocking is useful for diagnostics (connect Tx to Rx, and so on).

30. ATM adapter supports OAM
Recommended

Operation and maintenance (OAM) is needed for diagnostics.

This capability is recommended for Client systems, but is required for a Server system (layers F1–F5).

31. ATM adapter supports buffer chaining (Tx + Rx)
Recommended

This feature is needed for large packets.

This capability is recommended for Client systems, but is required for a Server system.

ADSL Requirements

New support is provided in the Windows and Windows NT operating systems for ADSL adapters, which provide a faster method for moving data over regular phone lines.

It is recommended that manufacturers participate in developing standards for this technology. For example, review the white paper, *An Interoperable End-to-End Broadband Service Architecture over ADSL Systems*, which discusses end-to-end service interoperability over ATM over ADSL. This paper, available at http://www.microsoft.com/isp/supercomm, was jointly developed by leading ADSL vendors.

32. ADSL device meets PC 98 network adapter requirements
Required

For PC 98, the driver must support NDIS 5.0. It is also recommended that the manufacturer participate in developing standards for ATM solutions.

33. ATM/ADSL solution is implemented

Recommended

Refer to ATM Adapter requirements for ATM specific requirements if an ATM/ADSL solution is implemented.

34. ADSL device supports RADSL

Recommended

On a rate adaptive digital subscriber line (RADSL), the downstream and upstream data rates are independently set either by an automatic adaptive algorithm or by manual selection.

RADSL provides the capability to optimize the transmission speed and performance over a range of telephone-line loop distances. Adaptive channel equalization ensures more robust performance in the presence of channel impairments and narrow-band interference.

This also helps telephone companies to provision RADSL access on their existing networks. RADSL products can be provisioned on many telephone lines without costly and time-consuming network upgrades.

PC 98 Design for Network Communications

This section summarizes requirements related to the PC 98 design initiatives defined in Part 1 of this guide.

Plug and Play and Bus Design for Network Communications

The items in this section are PC 98 requirements for Plug and Play capabilities.

35. Each device has a unique Plug and Play device ID

Required

For a system-board device, there must be a Plug and Play device-specific ID.

Each bus-specific device must provide Plug and Play device IDs in the manner required for the bus it uses, as defined in Part 3 of this guide. For example, a PCI device must comply with PCI 2.1 requirements and also must provide a Subsystem ID and Subsystem Vendor ID as defined in the “PCI” chapter in Part 3 of this guide.

36. Automatic resource assignment and dynamic disable capabilities are supported

Required

The system must be capable of automatically assigning, disabling, and relocating the resources used by this device as necessary using the method required for the related bus class. When an end user changes this device or adds it to the system, setting resource assignments must not require changing jumpers or switches on either the adapter or the system board. In the event of an irreconcilable conflict with other devices on the system, the system must be able to disable the device to prevent the system from stalling.

37. Plug and Play capabilities support multiple adapters

Required

For network communications devices, the Plug and Play IDs and resource support must be sufficient to automatically support the addition of multiple network communications devices to the system. This is true for the same and differing types of network communications devices.

38. All resource settings are reported in the user interface

Required

All resource settings must be viewable in Device Manager and in adapter properties dialog boxes. All resource settings that can be changed by the user must be changed using the standard Windows user interface, not by way of INI files or other setting files.

Power Management for Network Communications

This section summarizes the specific power management requirements for network communications devices.

39. Device complies with device class power management reference specification

Required

The *Network Device Class Power Management Reference Specification, Version 1.0* or higher, provides definitions of the OnNow device power states (D0–D3) for network adapters. The specification also covers device functionality expected in each power state and the possible wake-up event definitions for the class.

40. Device supports wake-up events

Required

This requirement applies specifically to Ethernet and Token Ring adapters. The *Network Device Class Power Management Reference Specification* does not support ATM and ISDN adapters.

The system must be capable of being awakened from a lower power state based on network events specified by the local networking software. This capability yields the result that any standard Windows network access—such as connections to shared drives and WinSock connections, as well as focused service and management applications—has the capability to wake machines in lower power states.

For PC 98, support is required for peer-to-peer networking, personal web servers, and other transparent networking applications. Wake-up capabilities must be based on pattern matching, which is a method of filtering, in addition to the normal address filtering that occurs when the system is fully on.

Implementation details are described in the “Network Wake-up Frames” and “Network Wake-up Frame Details” sections of *Network Device Class Power Management Reference Specification, Version 1.0* or higher.

Device Drivers and Installation for Network Communications

This section summarizes requirements for network communications device drivers, in addition to the requirements for using an NDIS 5.0 miniport driver as defined in the “System Requirements for Network Communications” section earlier in this chapter.

41. Device drivers and installation meet PC 98 requirements

Required

The manufacturer does not need to supply a driver if a PC 98-compliant driver provided with the operating system can be used. If the manufacturer supplies a driver, the requirements for the device drivers and installation are defined in the “Basic PC 98” chapter in Part 2 of this guide. The basic requirements include driver support for unattended installation and Help file support if special driver parameters are used.

For exceptions to unattended installation requirements for ISDN adapters, see the “ISDN Requirements” section earlier in this chapter.

For information about NDIS status codes and indication mechanisms, see the Microsoft Windows NT 5.0 DDK.

42. Driver supports promiscuous mode

Required

This ensures that the adapter can be used with Microsoft Network Monitor Agent. This requirement applies only to LAN (non-switched) media.

Notice that, by default, promiscuous mode is not turned on. Configuring promiscuous mode should be possible only by using the Microsoft Network Monitor Agent or another similar administrative application.

43. Driver works correctly with Microsoft network clients and protocols

Required

This includes the 32-bit Microsoft client and NetWare-compatible clients provided with Windows, whether connected to a Windows NT-based server, a Novell NetWare 3.*x* or 4.*x* server, or a Windows-based peer server. In all cases, this includes connections using Microsoft TCP/IP, IPX/SPX-compatible protocol, and NetBEUI.

44. NDIS miniport driver does not make operating system–specific kernel calls

Required

A miniport driver that follows the NDIS 5.0 specification must not make operating system–specific calls. A correct driver makes calls only to the NDIS library. The NDIS library provides all the functions a driver needs or should use. This results in binary compatibility of the driver between Windows and Windows NT.

NDIS conformance must be validated over a single network connection and multiple connections. For Windows NT, this must be validated on a multiprocessor system as part of PC 98 testing.

45. NDIS 5.0 driver uses new INF format

Required

For NDIS 5.0 drivers (which are required for Windows NT 5.0), all network components must use the new-style INF format, which is based on the Windows 95 INF format. For information, see the Windows NT 5.0 DDK.

Note: For Windows NT 5.0, there will be no legacy INF support and no satisfactory upgrade option for OEM components created for Windows NT 4.0.

Network Communications References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

1997 Version of National ISDN Basic Rate Interface Terminal Equipment Generic Guidelines, Document Number SR-3888
Phone: (800) 521-2673 (North America)
(908) 699-5800 (Outside North America)
http://www.bellcore.com

ATM User-Network Interface Specification, Version 3.1
Prentice Hall, 1995
ISBN 0-13-393828-X

Device Bay Interface Specification, Version 1.0
http://www.device-bay.org

Network Device Class Power Management Reference Specification, Version 1.0
http://www.microsoft.com/hwdev/onnow.htm

High-Speed Cable Data Service (HSCDS) Request for Proposals (RFP)
Cable Television Laboratories, Inc.
400 Centennial Parkway
Louisville, CO 80027
http://www.cablelabs.com

An Interoperable End-to-End Broadband Service Architecture over ADSL System
http://www.microsoft.com/isp/supercomm/ (click the "High Speed Access" link in the frame on the left, then scroll down to the bottom of the frame on the right)

NDIS and Windows networking white papers
http://www.microsoft.com/ntserver/communications/

USB specifications
Phone: (503) 264-0590
Fax: (503) 693-7975
http://www.usb.org

Windows 95 and Windows NT DDKs
MSDN Professional membership
Includes NDIS specifications, driver requirements, and Plug and Play support.

Checklist for Network Communications

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

Consumer PC 98	Office PC 98	Entertainment PC 98
1. PC system includes network adapter		
Recommended	*Required, if no modem*	*Recommended*
2. PC system includes internal or external ISDN device		
Recommended	*Recommended*	*Recommended*
3. PC system includes cable modem		
Optional	*Optional*	*Optional*
4. PC system includes ATM adapter		
Optional	*Optional*	*Optional*
5. PC system includes ADSL adapter		
Optional	*Optional*	*Optional*
6. PC system includes satellite or broadcast receiver with NDIS driver		
Recommended	*Recommended*	*Recommended*
7. Adapter uses NDIS 5.0 miniport driver		
Required		
8. Full-duplex adapter automatically detects and switches to duplex mode		
Required		
9. Adapter automatically senses presence of functional network		
Required		
10. Adapter automatically senses transceiver type		
Required		
11. Adapter supports quadword buffer alignment for receive and byte buffer alignment for send		
Required		
12. Adapter communicates with driver across any bridge		
Required		
13. Adapter supports filtering for 32 multicast addresses, at minimum		
Required		
14. Adapter is compatible with remote new system setup capabilities if used as boot device		
Required		
15. Device Bay network adapter meets PC 98 requirements		
Required		
16. Internal ISDN device meets PC 98 network adapter requirements		
Required		
17. Internal ISDN device supports synchronous HDLC framing		
Required		
18. Internal ISDN device uses NDIS WAN miniport driver		
Required		

19. Internal ISDN device includes connection for analog phone
Recommended

20. ISDN device supports auto-SPID detection algorithms and standard SPID format
Required

21. ISDN driver supports switch detection
Recommended

22. ISDN driver supports unattended installation, with limitations
Required

23. ISDN device includes built-in NT-1
Recommended

24. ATM adapter meets PC 98 network adapter requirements
Required

25. ATM adapter supports a minimum number of active connections
Required

26. ATM adapter supports all service types defined by the ATM Forum
Recommended

27. ATM adapter supports a minimum number of simultaneously active rt-VBR/nrt-VBR/CBR connections
Required

28. ATM adapter supports traffic shaping
Required

29. ATM adapter supports external clocking
Required

30. ATM adapter supports OAM
Recommended

31. ATM adapter supports buffer chaining (Tx + Rx)
Recommended

32. ADSL device meets PC 98 network adapter requirements
Required

33. ATM/ADSL solution is implemented
Recommended

34. ADSL device supports RADSL
Recommended

35. Each device has a unique Plug and Play device ID
Required

36. Automatic resource assignment and dynamic disable capabilities are supported
Required

37. Plug and Play capabilities support multiple adapters
Required

38. All resource settings are reported in the user interface
Required

39. Device complies with device class power management reference specification
Required

40. Device supports wake-up events
Required

41. Device drivers and installation meet PC 98 requirements
Required

42. Driver supports promiscuous mode
Required

43. Driver works correctly with Microsoft network clients and protocols
Required

44. NDIS miniport driver does not make operating system–specific kernel calls
Required

45. NDIS 5.0 driver uses new INF format
Required

C H A P T E R   2 1

Printers

This chapter presents the PC 98 requirements and recommendations for printers. Printers and other devices attached to parallel ports should be capable of high-speed, bi-directional data transfers. The design criteria for parallel devices follows the design criteria for parallel ports as described in the “I/O Ports and Devices” chapter in Part 4 of this guide.

The goal of the PC 98 requirements for printers and parallel ports is to ensure the following:

- Maximum speed for transfer of parallel data between the system and the peripheral
- A true Plug and Play experience for users

Contents

Basic Printer Features

This section summarizes the basic PC 98 hardware requirements for printers.

1. IEEE 1394 printer meets PC 98 requirements for IEEE 1394
Required

The IEEE 1394 bus is recommended for support of fast, high-density data transfer. For information about implementing IEEE 1394 for PC 98, see the "IEEE 1394" chapter in Part 3 of this guide.

2. USB printer meets PC 98 requirements for USB devices
Recommended

The USB bus is a requirement for PC 98 systems. It is recommended that USB printers conform to the *Universal Serial Bus Device Class Definition for Printing Devices, Version 1.0* or higher. For information about implementing USB for PC 98, see the "USB" chapter in Part 3 of this guide.

3. IEEE 1284 printer supports compatibility mode, nibble mode, and ECP, compliant with IEEE 1284-I
Required

Parallel peripherals must implement nibble mode and compatibility mode. Nibble mode provides a means of transferring the identification string from the peripheral to the system. Compatibility mode provides backward compatibility with non-Plug and Play systems that do not support more advanced modes.

A parallel device complies with IEEE 1284 if it meets the required criteria documented in the IEEE 1284 specification, *Standard Signaling Method for a Bi-directional Parallel Peripheral Interface for Personal Computers.* For a parallel device that connects to a PC 98 system, the minimum requirement is IEEE 1284 Level I compliance, which implements the compatibility and nibble modes as specified in IEEE 1284 and defines the mechanical and electrical specifications of the peripheral.

An IEEE 1284-I–compliant peripheral uses the standard IEEE 1284-B connector. In all cases, ensure that there is enough space between the connectors and the surrounding enclosure to allow for a mating connector, a connector shell, and a latch assembly.

For more information about the electrical specifications for IEEE 1284-I–compliant peripherals, refer to the IEEE 1284 specification.

For more information, see the following related parallel port requirements defined in the "I/O Ports and Devices" chapter in Part 4 of this guide:

- Support for compatibility, nibble mode, and ECP protocols compliant with IEEE 1284-1994
- Port connectors compliant with IEEE 1284-I, at minimum
- Support for ECP mode compliant with IEEE 1284

4. IEEE 1284 printer meets IEEE 1284-II requirements

Recommended

Peripheral devices capable of handling a high-speed data rate should comply with the mechanical, electrical, and protocol specification of IEEE 1284-II. In particular, such devices should support the protocols of the IEEE 1284-II ECP mode and should use the IEEE 1284-C connector.

5. ECP printer works correctly when ECP mode is turned off

Required

This ensures that the user has correct printing support when ECP mode is not in use.

6. IEEE 1284 hardware supports error notification

Required

The following minimum errors must be reported individually by the hardware:

- Out of paper
- Paper jam
- Load other paper size

PC 98 Printer Design

This section summarizes requirements related to the PC 98 design initiatives in Part 1 of this guide.

Plug and Play for Printers

The items in this section are requirements for Plug and Play capabilities. For Plug and Play requirements related to the printer port on the PC, see the "I/O Ports and Devices" chapter in Part 4 of this guide or the related bus port requirements in Part 3 of this guide.

7. Implement Plug and Play support for all supported buses
Required

Complete Plug and Play support must be implemented for all buses that the device supports. For information about the Plug and Play requirements, see the related bus-class definitions in Part 3 of this guide.

8. Peripheral device meets IEEE 1284 requirements
Required

Recommended: Support CompatibleID key in the device identification string.

These requirements include a Plug and Play device ID as described in the IEEE 1284 specification. For more information, see the "I/O Ports and Devices" chapter in Part 4 of this guide.

Device Drivers and Installation for Printers

This section summarizes device driver requirements for printers. The items in this section are requirements for all PC 98 systems.

9. Printer INF file and installation meet PC 98 requirements
Required

Each device requires a printer INF file for both Windows and Windows NT operating systems. The manufacturer does not need to supply a printer INF file if a standard printer INF file provided with the operating system can be used.

If the manufacturer provides an INF file, it must be complete and free of errors. This INF file must comply with the printer-specific extensions listed in the Windows and Windows NT DDKs.

If the manufacturer supplies an INF file or another file, the requirements for printer INF files and installation include the following:

- All devices and files must pass PC 98 compliance testing
- All configuration settings are stored in the registry
- The correct files specified in the device's INF file must be installed in the correct locations
- Driver installation and removal use Windows-based methods as defined in the Windows and Windows NT DDKs
- Files provided by the vendor must not use the same file names used by files included in Windows operating systems unless specifically agreed upon with Microsoft

For complete details about standard installation requirements for device drivers, see the "Basic PC 98" chapter in Part 2 of this guide.

10. Driver correctly reports device capabilities
Required
For Windows, this means that the driver correctly supports the DEVMODE structure as defined in the Windows and Windows NT DDKs.

11. Driver supports error notification
Required
At a minimum, the device driver must support notifying the user of errors reported by the hardware.

12. Driver supports ICC color matching
Required
Windows and Windows NT support using color profiles that comply with the ICC Profile Format specification. For contact information on device profiles, see the references at the end of this chapter. The ICM APIs and functionality for Windows and Windows NT are described in the Win32 SDK and the Windows NT 5.0 DDK.

For PC 98, color-capable devices such as desktop monitors, printers, scanners, still-image cameras, LCDs, color plasma displays, or other flat-panel devices are required to install one or more ICC profiles for ICM. Providing a monitor color-calibration utility is recommended for generating, editing, and installing ICC profiles. The sRGB profile will be distributed in Windows and Windows NT.

13. Port monitor software meets DDK guidelines

Required

If the device includes bi-directional port monitor software that replaces the default Windows port monitor, then this software must accurately report errors. For information about implementing port monitor software, see the Windows and Windows NT DDKs.

14. Driver supports point-and-print network installation

Required

This means that the user is not required to provide disks or files when installing a new printer of the same type as another printer already on the network.

15. Device available immediately following installation

Required

The user should not have to restart the system immediately after device installation in order to print.

16. Device supports accurate printable regions

Required

The printable regions that can be selected in the user interface must be accurately supported in the actual print output.

17. Driver supports required DDIs

Required

For Windows NT drivers, the device driver interfaces (DDIs) are defined in the Windows NT DDK. Win32-based printer drivers ensure that print commands from Windows NT-based applications are executed correctly on the specified printer or plotter. Because Win32 APIs are not hardware-specific, it is the job of each printer driver to interpret the commands for its specific hardware.

For Windows drivers, this requirement includes correct support of all features advertised for the device, plus required support for Windows features. The DDIs that must be supported are listed in the “Printer Driver Overview” section of the Windows DDK. This includes the following support, in addition to other support defined in the DDK:

- TrueType glyph indexes
- Big fonts (those that require more than 64K to express)
- Enhanced metafile (EMF) spooling
- Bezier curve output
- Services from the Windows device-independent bitmap (DIB) engine

18. Driver based on unidriver

Recommended

Microsoft provides a universal printer driver (unidriver), which is capable of carrying out requests (such as printing text, rendering bitmaps, or advancing a page) on most printer types. To build a driver for a particular printer, a developer builds a minidriver. This minidriver accepts requests from the GDI and then, in most cases, passes the request to the unidriver along with information that describes the capabilities, commands, and resident fonts of the particular printer. For more information, see the Windows NT and Windows DDKs.

Printer References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

ICC Profile Format Specification, Version 3.4,
International Color Consortium
http://www.color.org

*Standard Signaling Method for a Bi-directional Parallel Peripheral Interface for Personal Computers (*IEEE 1284 specification)
ASK*IEEE
Phone: (800) 949-4333
Fax: (212) 310-4091

Global Engineering Documents
Phone: (800) 854-7179 (US)
(613) 237-4250 (Canada)
(303) 792-2181 (Outside North America)
Fax: (303) 397-2740
ftp://ftp.symbios.com/pub/standards/io/

Universal Serial Bus Device Class Definition for Printing Devices, Version 1.0
Phone: (503) 264-0590
Fax: (503) 693-7975
http://www.usb.org

Windows and Windows NT DDKs and Win32 SDK
MSDN Professional membership

Checklist for Printers

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

1. IEEE 1394 printer meets PC 98 requirements for IEEE 1394
Required

2. USB printer meets PC 98 requirements for USB devices
Recommended

3. IEEE 1284 printer supports compatibility mode, nibble mode, and ECP, compliant with IEEE 1284-I
Required

4. IEEE 1284 printer meets IEEE 1284-II requirements
Recommended

5. ECP printer works correctly when ECP mode is turned off
Required

6. IEEE 1284 hardware supports error notification
Required

7. Implement Plug and Play support for all supported buses
Required

8. Peripheral device meets IEEE 1284 requirements
Required

9. Printer INF file and installation meet PC 98 requirements
Required

10. Driver correctly reports device capabilities
Required

11. Driver supports error notification
Required

12. Driver supports ICC color matching
Required

13. Port monitor software meets DDK guidelines
Required

14. Driver supports point-and-print network installation
Required

15. Device available immediately following installation
Required

16. Device supports accurate printable regions
Required

17. Driver supports required DDIs
Required

18. Driver based on unidriver
Recommended

C H A P T E R 2 2

Scanners and Digital Cameras

22

This chapter presents the PC 98 requirements and recommendations for still-image devices, specifically digital cameras and scanners, including sheetfed, flatbed, handheld, and film-scanning devices.

The still-image device market is growing rapidly in the consumer and mass-market segments, stimulated by applications such as Internet authoring and home publishing, and by the availability of low-cost, high-resolution color printers. For PC 98, the important design issues include the following:

- Increased image-resolution quality at consumer price points
- Incorporation of new bus standards such as USB and IEEE 1394 into still-image peripheral devices
- Integration of a fast transfer mechanism for non-tethered devices such as digital cameras
- Improved ease of connectivity and installation for the end user—that is, by reducing components to be installed, such as power supplies and add-on cards, by simplifying the installation procedure to limit the number of user responses required, and by ensuring compliance with Plug and Play requirements
- Incorporation of color-calibration support across devices
- Implementation of push-model behavior, where events are triggered based on user-initiated actions at the device, such as inserting paper or pushing a button

Contents

Scanner and Digital Camera Basic Features

This section summarizes the basic PC 98 hardware requirements for scanners and digital cameras.

1. Device uses PC 98 compatible port connection
Required

Recommended: IEEE 1394 connection.

No legacy or proprietary solutions are acceptable for PC 98.

2. Icons provided for port and peripheral connectors
Required

To ensure proper connection by the user between cable and connector, an icon or text identifier must be added to any external connector, using vendor designs or the icons provided in the "Icons" appendix in the References part of this guide. The icon can be molded into or printed on the plastic (either by stamping or by affixing as a permanent sticker).

3. Device supports ICC color matching
Required

Windows and Windows NT support using color profiles that comply with the ICC Profile Format specification. All color output from still-image devices must be defined. The device either must create sRGB output or must embed the ICC profile for the newly acquired image into the image file to identify the color-space information for that image.

For contact information on device profiles, see the references at the end of this chapter. The ICM APIs and functionality for Windows and Windows NT are described in the Win32 SDK and the Windows NT 5.0 DDK.

For PC 98, color-capable devices such as desktop monitors, printers, scanners, still-image cameras, LCDs, color plasma displays, or other flat-panel devices are required to install one or more ICC profiles for ICM. Providing a monitor color-calibration utility is recommended for generating, editing, and installing ICC profiles. The sRGB profile will be distributed in Windows and Windows NT.

4. IR device meets PC 98 IR requirements
Required

For imaging devices that include an IR interface, all IR hardware must at minimum comply with the IR requirements defined in the "I/O Ports and Devices" chapter in Part 3 of this guide.

5. SCSI device meets PC 98 SCSI requirements

Required

All SCSI hardware must comply with the requirements defined in the “SCSI” chapter in Part 3 of this guide. This ensures complete Plug and Play capabilities with SCSI hardware. For example, a user must be able to attach any SCSI peripheral on a system with SCSI support. The operating system should automatically recognize it, load and initialize the appropriate drivers, and then make the device available for use.

6. SCSI device attaches to any PC 98-compliant SCSI controller

Required

All SCSI scanners must be able to attach successfully to any SCSI controller that meets the PC 98 requirements defined in the “SCSI” chapter in Part 3 of this guide.

7. USB device meets PC 98 USB requirements

Required

All USB hardware must comply with the requirements defined in the “USB” chapter in Part 3 of this guide, which includes the USB specifications for specific device types. This ensures complete Plug and Play capabilities with USB hardware and meets all the core and device requirements for USB. For example, a user must be able to dynamically attach any USB peripheral to any USB connector. The operating system should automatically recognize the device, load and initialize the appropriate drivers, and then make the device available for use.

All devices must comply with the requirements defined in the *USB Imaging Class Specification*, and drivers must be implemented under the WDM Still Image architecture, as defined in the Windows NT 5.0 DDK.

8. USB device supports string descriptors

Required

The device descriptor, as listed in Section 9.6.1 of the USB specification, must have valid iManufacturer and iProduct string descriptor indexes. All USB scanners must comply with requirements defined in Sections 9.4.3 and 9.6.5 of the USB specification.

9. IEEE 1394 device meets PC 98 requirements for IEEE 1394

Required

All IEEE 1394 hardware must comply with the requirements defined in the “IEEE 1394” chapter in Part 3 of this guide.

PC 98 Design for Scanners and Digital Cameras

This section summarizes requirements related to the PC 98 design initiatives described in Part 1 of this guide.

Plug and Play for Scanners and Digital Cameras

The items in this section are requirements for Plug and Play capabilities. For Plug and Play requirements related to parallel ports, see the "I/O Ports and Devices" chapter in Part 4 of this guide or the related bus port requirements in Part 3 of this guide.

10. Plug and Play capabilities implemented for all supported buses
Required

Complete Plug and Play capabilities must be implemented for all buses that the device supports. For information about the Plug and Play requirements, see the related bus requirements in Part 3 of this guide.

11. Each device has a Plug and Play device ID
Required

All devices for all buses must supply a human-readable device ID in the manner required for the bus it uses. The device ID requirements for each bus type are defined in Part 3 of this guide; however, the device ID requirements for devices that use parallel ports are defined in the IEEE 1284 specification, as summarized in the "I/O Ports and Devices" chapter in Part 4 of this guide.

Scanner and Digital Camera Power Management

This section summarizes the specific power management requirements for scanners and digital cameras.

12. Device supports power management requirements for its bus
Required

The device must support the power management requirements for the bus it uses, as defined in Part 3 of this guide.

Device Drivers and Installation for Scanners and Digital Cameras

This section summarizes the device driver requirements for scanners and digital cameras.

13. Device drivers and installation meet PC 98 requirements

Required

The manufacturer does not need to supply a driver if a PC 98-compliant driver provided with the operating system can be used. If the manufacturer supplies a driver, the requirements for the device drivers and installation are defined in the "Basic PC 98" chapter in Part 2 of this guide. The basic requirements include driver support for unattended installation and Help file support if special driver parameters are used.

14. Driver support is implemented under Still Image architecture

Required

Still image devices such as scanners and digital cameras must use the Still Image architecture based on WDM. Still digital cameras capable of creating video streams also must provide a WDM minidriver based on WDM Stream class support.

For information about Still Image architecture and WDM support, see the Windows NT 5.0 DDK. See also the related articles on the web site at http://www.microsoft.com/hwdev/pcfuture/.

15. Applications provided with the device meet Win32 specifications

Required

Any Windows-based applications provided with the device must meet Microsoft requirements for software compatibility as defined in the Win32 SDK.

Scanner and Digital Camera References

The following represents some of the references, services, and tools available to help build hardware that is optimized to work with Windows operating systems.

Device class power management reference specifications
http://www.microsoft.com/hwdev/onnow.htm

International Color Consortium (ICC)
ICC Profile Format Specification
http://www.color.org

Plug and Play specifications
http://www.microsoft.com/hwdev/specs/

Universal Serial Bus Specification, Version 1.0
USB Imaging Class Specification
Phone: (503) 264-0590
Fax: (503) 693-7975
http://www.usb.org

WDM device driver support and WDM Still Image architecture white papers
http://www.microsoft.com/hwdev/pcfuture/

Windows and Windows NT DDKs and Win32 SDK
MSDN Professional membership

Checklist for Scanners and Digital Cameras

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

1. Device uses PC 98 compatible port connection
Required

2. Icons provided for port and peripheral connectors
Required

3. Device supports ICC color matching
Required

4. IR device meets PC 98 IR requirements
Required

5. SCSI device meets PC 98 SCSI requirements
Required

6. SCSI device attaches to any PC 98-compliant SCSI controller
Required

7. USB device meets PC 98 USB requirements
Required

8. USB device supports string descriptors
Required

9. IEEE 1394 device meets PC 98 requirements for IEEE 1394
Required

10. Plug and Play capabilities implemented for all supported buses
Required

11. Each device has a Plug and Play device ID
Required

12. Device supports power management requirements for its bus
Required

13. Device drivers and installation meet PC 98 requirements
Required

14. Driver support is implemented under Still Image architecture
Required

15. Applications provided with the device meet Win32 specifications
Required

References

A P P E N D I X   A

Icons

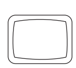
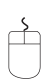
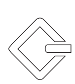
This appendix presents a recommended set of standard icons that can be used for connectors on a personal computer. A set of icons must be added to any external connector on a PC 98 system. And the same icons also should be added to any cable connectors that plug into the PC. (Icons are not required for peripherals or the peripheral end of the cable.)

Although no specific types of icons are required, you can base your icons either on existing vendor designs or on the designs shown in this appendix. Files that contain art for these icons, plus alternative designs from Hewlett-Packard, are available at http://www.microsoft.com/desguid/icons.htm.

For PC cases and cable-plug housings, the icon can be molded into the plastic. Icons printed on permanent labels are also acceptable.

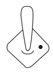
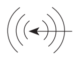
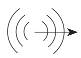
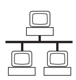
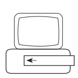
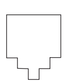
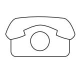
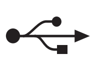
PC 98 does not specifically require or recommend color coding of connectors and other cable markings, but the PC designer is encouraged to implement color coding in order to enhance user accessibility.

Connector Icons

Suggested icons	Description
	Monitor
	Keyboard
	Mouse
	SCSI

Continued

Connector Icons *(continued)*

Suggested icons	Description
	Parallel/Printer
	Gameport/Joystick
	Monaural/Stereo In
	Monaural/Stereo Out
	Microphone
	Serial Port
	Serial Port 1
	Serial Port 2
	Network/Thicknet+Twisted
	Headphone
	Expansion Bus/Docking Station
	Telephone Line
	Telephone Set
	USB

A P P E N D I X   B

Device Identifiers

This appendix lists CompatibleIDs for Plug and Play vendor IDs and device IDs.

Note: For non-BIOS enumerated Industry Standard Architecture (ISA) devices, new vendor IDs must be registered by sending e-mail to pnpid@microsoft.com.

Contents

Plug and Play Vendor and Device IDs

All non-BIOS enumerated devices must not use “PNP” in their vendor and device codes. Instead, the vendor must register with Extended Industry Standard Architecture (EISA) and have a vendor code assigned (for example, CTL for Creative Labs). The PNP vendor code is reserved for Microsoft and can be used only when defining a device’s CompatibleID after indicating the device’s HardwareID in the Plug and Play header.

Use of CompatibleIDs is strongly recommended for devices that use inbox device drivers, such as a “Standard PC COM Port” (PNP0500) or “Sound Blaster 16-compatible Sound Device” (PNPB003).

The following example output of a Plug and Play header from Isolate.exe is provided as a reference for the Microsoft Windows operating system.

```
Vendor ID:     XXXFFFF
Serial Number: 00000001
Checksum (reported): 0x5E
PNP Version:   1.0
Vendor Ver.:   10
Device Description: IDE Port
Device ID:  XXX0001
Doesn't Support I/O Range Checking
Vendor Defined Logical Device Control Registers:
None
Compatible Device ID: PNP0600
Device Description: IDE

Dependent Function 0
⋮
Dependent Function 1
⋮
End of Dependent Functions
```

When the user is installing devices that use this method, a dialog box appears at the beginning of the enumeration sequence to suggest use of the Windows 95 default driver. Windows 95 also provides the option of using a manufacturer-supplied disk in case the user wants to choose a manufacturer-supplied driver.

For multifunction adapters, you should supply an INF file that chooses the appropriate drivers (including default drivers) for all the adapter’s devices. This prevents additional dialog boxes from repeatedly requesting the default driver or a manufacturer’s disk for the remaining devices on the adapter.

When an INF file is used in this manner for default driver selection, it must link the HardwareID (XXX0000) to the appropriate compatible device driver from the Windows 95 distribution CD or installation disks. If this is not done, Windows 95 will continue to query the user for either the default driver or a new driver, thus defeating the purpose of using the INF file in this way.

Generic Windows Device IDs

Many devices, such as the interrupt controller or the keyboard controller, have no standard EISA ID. Also, a set of compatible devices, such as video graphics array (VGA) and Super VGA (SVGA), are not actually devices but define a compatibility hardware subset. Yet another set of IDs needs to be used to identify buses.

Microsoft has reserved an EISA prefix (PNP) to identify various devices that do not have existing EISA IDs. Microsoft also uses PNP to define compatibility devices. The IDs are defined in the following tables.

Device ID Ranges

ID range	Category
PNP0xxx	System devices
PNP8xxx	Network adapters
PNPAxxx	Small computer system interface (SCSI), proprietary CD adapters
PNPBxxx	Sound, video capture, multimedia
PNPCxxx–Dxxx	Modems

The following obsolete device ID is provided only for compatibility with earlier device ID lists.

Device ID	Description
PNP0802	Microsoft Sound System-compatible device (obsolete; use PNPB0xx instead)

Interrupt Controllers

Device ID	Description
PNP0000	AT interrupt controller
PNP0001	EISA interrupt controller
PNP0002	MCA interrupt controller
PNP0003	Advanced Protocol Interrupt Controller (APIC)
PNP0004	Cyrix SLiC MP interrupt controller

Timers

Device ID	Description
PNP0100	AT timer
PNP0101	EISA timer
PNP0102	MCA timer

DMA

Device ID	Description
PNP0200	AT direct memory access (DMA) controller
PNP0201	EISA DMA controller
PNP0202	MCA DMA controller

Keyboards

Device ID	Description
PNP0300	IBM PC/XT keyboard controller (83-key)
PNP0301	IBM PC/AT keyboard controller (86-key)
PNP0302	IBM PC/XT keyboard controller (84-key)
PNP0303	IBM Enhanced (101/102-key, PS/2 mouse support)
PNP0304	Olivetti keyboard (83-key)
PNP0305	Olivetti keyboard (102-key)
PNP0306	Olivetti keyboard (86-key)
PNP0307	Microsoft Windows keyboard
PNP0308	General Input Device Emulation Interface (GIDEI) legacy

Continued

Keyboards *(continued)*

Device ID	Description
PNP0309	Olivetti keyboard (A101/102-key)
PNP030A	AT&T 302 keyboard
PNP030B	Reserved by Microsoft
PNP0320	Japanese keyboard A01 (106-key)
PNP0321	Japanese keyboard (101-key)
PNP0322	Japanese AX keyboard
PNP0323	Japanese keyboard 002/003 (106-key)
PNP0324	Japanese keyboard 001 (106-key)
PNP0325	Japanese Toshiba desktop keyboard
PNP0326	Japanese Toshiba laptop keyboard
PNP0327	Japanese Toshiba notebook keyboard
PNP0340	Korean keyboard (84-key)
PNP0341	Korean keyboard (86-key)
PNP0342	Korean enhanced keyboard
PNP0343	Korean enhanced keyboard 101b
PNP0343	Korean enhanced keyboard 101c
PNP0344	Korean enhanced keyboard 103

Parallel Devices

Device ID	Description
PNP0400	Standard LPT port
PNP0401	Extended capabilities port (ECP) printer port

Serial Devices

Device ID	Description
PNP0500	Standard PC COM port
PNP0501	16550A-compatible COM port
PNP0510	Generic IrDA-compatible device

Disk Controllers

Device ID	Description
PNP0600	Generic ESDI/IDE/ATA-compatible hard disk controller
PNP0601	Plus Hardcard II
PNP0602	Plus Hardcard IIXL/EZ
PNP0603	Generic Integrated Device Electronics (IDE) supporting Device Bay specifications
PNP0700	PC standard floppy disk controller (FDC)
PNP0701	Standard FDC supporting Device Bay specification

Display Adapters

Device ID	Description
PNP0900	VGA compatible
PNP0901	Video Seven VRAM/VRAM II/1024i
PNP0902	8514/A compatible
PNP0903	Trident VGA
PNP0904	Cirrus Logic laptop VGA
PNP0905	Cirrus Logic VGA
PNP0906	Tseng ET4000
PNP0907	Western Digital VGA
PNP0908	Western Digital laptop VGA
PNP0909	S3 Inc. 911/924
PNP090A	ATI Ultra Pro/Plus (Mach 32)
PNP090B	ATI Ultra (Mach 8)
PNP090C	XGA compatible
PNP090D	ATI VGA Wonder
PNP090E	Weitek P9000 graphics adapter
PNP090F	Oak Technology VGA
PNP0910	Compaq QVision
PNP0911	XGA/2
PNP0912	Tseng Labs W32/W32i/W32p
PNP0913	S3 Inc. 801/928/964
PNP0914	Cirrus Logic 5429/5434 (memory-mapped)

Continued

Display Adapters *(continued)*

Device ID	Description
PNP0915	Compaq Advanced VGA (AVGA)
PNP0916	ATI Ultra Pro Turbo (Mach 64)
PNP0917	Reserved by Microsoft
PNP0918	Matrox MGA
PNP0919	Compaq QVision 2000
PNP091A	Tseng W128
PNP0930	Chips & Technologies SVGA
PNP0931	Chips & Technologies Accelerator
PNP0940	NCR 77c22e SVGA
PNP0941	NCR 77c32blt
PNP09FF	Plug and Play monitors (VESA display data channel [DDC])

Peripheral Buses

Device ID	Description
PNP0A00	ISA bus
PNP0A01	EISA bus
PNP0A02	MCA bus
PNP0A03	Peripheral Component Interconnect (PCI) bus
PNP0A04	VESA/VL-bus
PNP0A05	Generic Advanced Configuration and Power Interface (ACPI) bus
PNP0A06	Generic ACPI Extended I/O (EIO) bus

Real-Time Clock, BIOS, and System Board Devices

Device ID	Description
PNP0800	AT-style speaker sound
PNP0B00	AT real-time clock
PNP0C00	Plug and Play BIOS (only created by the ROOT enumerator)
PNP0C01	System board
PNP0C02	General ID for reserving resources required by Plug and Play system board registers (not specific to a particular device)
PNP0C03	Plug and Play BIOS event notification interrupt

Continued

Real-Time Clock, BIOS, and System Board Devices *(continued)*

Device ID	Description
PNP0C04	Math co-processor
PNP0C05	Advanced Power Management (APM) BIOS (version-independent)
PNP0C06	Reserved for identification of early Plug and Play BIOS implementation
PNP0C07	Reserved for identification of early Plug and Play BIOS implementation
PNP0C08	ACPI system board hardware
PNP0C09	ACPI embedded controller
PNP0C0A	ACPI control method battery
PNP0C0B	ACPI fan
PNP0C0C	ACPI power-button device
PNP0C0D	ACPI lid device
PNP0C0E	ACPI sleep-button device
PNP0C0F	PCI interrupt link device
PNP0C10	ACPI system indicator device
PNP0C11	ACPI thermal zone
PNP0C12	Device Bay Controller (DBC)

PCMCIA Controller Chip Sets

Device ID	Description
PNP0E00	Intel 82365-compatible PCMCIA controller
PNP0E01	Cirrus Logic CL-PD6720 PCMCIA controller
PNP0E02	VLSI VL82C146 PCMCIA controller
PNP0E03	Intel 82365-compatible CardBus controller

Mouse

Device ID	Description
PNP0F00	Microsoft bus mouse
PNP0F01	Microsoft serial mouse
PNP0F02	Microsoft InPort mouse
PNP0F03	Microsoft PS/2-style mouse
PNP0F04	Mouse Systems mouse

Continued

Mouse *(continued)*

Device ID	Description
PNP0F05	Mouse Systems 3-button mouse (COM2)
PNP0F06	Genius mouse (COM1)
PNP0F07	Genius mouse (COM2)
PNP0F08	Logitech serial mouse
PNP0F09	Microsoft BallPoint serial mouse
PNP0F0A	Microsoft Plug and Play mouse
PNP0F0B	Microsoft Plug and Play BallPoint mouse
PNP0F0C	Microsoft-compatible serial mouse
PNP0F0D	Microsoft InPort-compatible mouse
PNP0F0E	Microsoft-compatible PS/2-style mouse
PNP0F0F	Microsoft Serial BallPoint-compatible mouse
PNP0F10	Texas Instruments QuickPort mouse
PNP0F11	Microsoft-compatible bus mouse
PNP0F12	Logitech PS/2-style mouse
PNP0F13 [1]	PS/2 port for PS/2-style mouse
PNP0F14	Microsoft Kids mouse
PNP0F15	Logitech bus mouse
PNP0F16	Logitech SWIFT device
PNP0F17	Logitech-compatible serial mouse
PNP0F18	Logitech-compatible bus mouse
PNP0F19	Logitech-compatible PS/2-style mouse
PNP0F1A	Logitech-compatible SWIFT device
PNP0F1B	HP Omnibook mouse
PNP0F1C	Compaq LTE Trackball PS/2-style mouse
PNP0F1D	Compaq LTE Trackball serial mouse
PNP0F1E	Microsoft Kids Trackball mouse
PNP0F1F	Reserved by Microsoft Input Device Group
PNP0F20	Reserved by Microsoft Input Device Group
PNP0F21	Reserved by Microsoft Input Device Group
PNP0F22	Reserved by Microsoft Input Device Group
PNP0F23	Reserved by Microsoft Input Device Group
PNP0FFF	Reserved by Microsoft Systems

[1] The system BIOS should report the PS/2 port, not which type of mouse is connected to that port.

Network Adapters

Device ID	Description
PNP8001	Novell/Anthem NE3200
PNP8004	Compaq NE3200
PNP8006	Intel EtherExpress/32
PNP8008	HP Ethertwist EISA LAN Adapter/32 (HP27248A)
PNP8065	Ungermann-Bass NIUps or NIUps/EOTP
PNP8072	DEC (DE211) Etherworks MC/TP
PNP8073	DEC (DE212) Etherworks MC/TP_BNC
PNP8078	DCA 10-MB MCA
PNP8074	HP MC LAN Adapter/16 TP (PC27246)
PNP80C9	IBM Token Ring
PNP80CA	IBM Token Ring II
PNP80CB	IBM Token Ring II/Short
PNP80CC	IBM Token Ring 4/16-MB
PNP80D3	Novell/Anthem NE1000
PNP80D4	Novell/Anthem NE2000
PNP80D5	NE1000 compatible
PNP80D6	NE2000 compatible
PNP80D7	Novell/Anthem NE1500T
PNP80D8	Novell/Anthem NE2100
PNP80DD	SMC ARCNETPC
PNP80DE	SMC ARCNET PC100, PC200
PNP80DF	SMC ARCNET PC110, PC210, PC250
PNP80E0	SMC ARCNET PC130/E
PNP80E1	SMC ARCNET PC120, PC220, PC260
PNP80E2	SMC ARCNET PC270/E
PNP80E5	SMC ARCNET PC600W, PC650W
PNP80E7	DEC DEPCA
PNP80E8	DEC (DE100) EtherWorks LC
PNP80E9	DEC (DE200) EtherWorks Turbo

Continued

Network Adapters *(continued)*

Device ID	Description
PNP80EA	DEC (DE101) EtherWorks LC/TP
PNP80EB	DEC (DE201) EtherWorks Turbo/TP
PNP80EC	DEC (DE202) EtherWorks Turbo/TP_BNC
PNP80ED	DEC (DE102) EtherWorks LC/TP_BNC
PNP80EE	DEC EE101 (built-in)
PNP80EF	DECpc 433 WS (built-in)
PNP80F1	3Com EtherLink Plus
PNP80F3	3Com EtherLink II or IITP (8-bit or 16-bit)
PNP80F4	3Com TokenLink
PNP80F6	3Com EtherLink 16
PNP80F7	3Com EtherLink III
PNP80F8	3Com generic EtherLink Plug and Play device
PNP80FB	Thomas-Conrad TC6045
PNP80FC	Thomas-Conrad TC6042
PNP80FD	Thomas-Conrad TC6142
PNP80FE	Thomas-Conrad TC6145
PNP80FF	Thomas-Conrad TC6242
PNP8100	Thomas-Conrad TC6245
PNP8105	DCA 10-MB
PNP8106	DCA 10-MB Fiber Optic
PNP8107	DCA 10-MB Twisted Pair
PNP8113	Racal NI6510
PNP811C	Ungermann-Bass NIUpc
PNP8120	Ungermann-Bass NIUpc/EOTP
PNP8123	SMC StarCard PLUS (WD/8003S)
PNP8124	SMC StarCard PLUS with on-board hub (WD/8003SH)
PNP8125	SMC EtherCard PLUS (WD/8003E)
PNP8126	SMC EtherCard PLUS with boot ROM socket (WD/8003EBT)
PNP8127	SMC EtherCard PLUS with boot ROM socket (WD/8003EB)

Continued

Network Adapters *(continued)*

Device ID	Description
PNP8128	SMC EtherCard PLUS TP (WD/8003WT)
PNP812A	SMC EtherCard PLUS 16 with boot ROM socket (WD/8013EBT)
PNP812D	Intel EtherExpress 16 or 16TP
PNP812F	Intel TokenExpress 16/4
PNP8130	Intel TokenExpress MCA 16/4
PNP8132	Intel EtherExpress 16 (MCA)
PNP8137	Artisoft AE-1
PNP8138	Artisoft AE-2 or AE-3
PNP8141	Amplicard AC 210/XT
PNP8142	Amplicard AC 210/AT
PNP814B	Everex SpeedLink /PC16 (EV2027)
PNP8155	HP PC LAN Adapter/8 TP (HP27245)
PNP8156	HP PC LAN Adapter/16 TP (HP27247A)
PNP8157	HP PC LAN Adapter/8 TL (HP27250)
PNP8158	HP PC LAN Adapter/16 TP Plus (HP27247B)
PNP8159	HP PC LAN Adapter/16 TL Plus (HP27252)
PNP815F	National Semiconductor Ethernode *16AT
PNP8160	National Semiconductor AT/LANTIC Ethernode 16-AT3
PNP816A	NCR Token-Ring 4-MB ISA
PNP816D	NCR Token-Ring 16/4-MB ISA
PNP8191	Olicom 16/4 Token Ring Adapter
PNP81C3	SMC EtherCard PLUS Elite (WD/8003EP)
PNP81C4	SMC EtherCard PLUS 10T (WD/8003W)
PNP81C5	SMC EtherCard PLUS Elite 16 (WD/8013EP)
PNP81C6	SMC EtherCard PLUS Elite 16T (WD/8013W)
PNP81C7	SMC EtherCard PLUS Elite 16 Combo (WD/8013EW or 8013EWC)
PNP81C8	SMC EtherElite Ultra 16
PNP81E4	Pure Data PDI9025-32 (Token Ring)
PNP81E6	Pure Data PDI508+ (ArcNet)
PNP81E7	Pure Data PDI516+ (ArcNet)
PNP81EB	Proteon Token Ring (P1390)
PNP81EC	Proteon Token Ring (P1392)

Continued

Network Adapters *(continued)*

Device ID	Description
PNP81ED	Proteon ISA Token Ring (1340)
PNP81EE	Proteon ISA Token Ring (1342)
PNP81EF	Proteon ISA Token Ring (1346)
PNP81F0	Proteon ISA Token Ring (1347)
PNP81FF	Cabletron E2000 Series DNI
PNP8200	Cabletron E2100 Series DNI
PNP8209	Zenith Data Systems Z-Note
PNP820A	Zenith Data Systems NE2000-compatible
PNP8213	Xircom Pocket Ethernet II
PNP8214	Xircom Pocket Ethernet I
PNP821D	RadiSys EXM-10
PNP8227	SMC 3000 Series
PNP8228	SMC 91C2 controller
PNP8231	Advanced Micro Devices AM2100/AM1500T
PNP8263	Tulip NCC-16
PNP8277	Exos 105
PNP828A	Intel 595-based Ethernet
PNP828B	TI2000-style Token Ring
PNP828C	AMD PCNet Family cards
PNP828D	AMD PCNet32 (VL-bus version)
PNP8294	IrDA Infrared NDIS driver (Microsoft-supplied)
PNP82BD	IBM PCMCIA-NIC
PNP82C2	Xircom CE10
PNP82C3	Xircom CEM2
PNP8321	DEC Ethernet (all types)
PNP8323	SMC EtherCard (all types except 8013/A)
PNP8324	ARCNET-compatible
PNP8326	Thomas Conrad (all ARCNET types)
PNP8327	IBM Token Ring (all types)
PNP8385	Remote network access (RNA) driver
PNP8387	RNA point-to-point protocol (PPP) driver
PNP8388	Reserved for Microsoft networking components
PNP8389	Peer IrLAN IR driver (Microsoft-supplied)

SCSI and Proprietary CD-ROM Adapters

Device ID	Description
PNPA002	Future Domain 16-700–compatible controller
PNPA003	Panasonic proprietary CD-ROM adapter (SBPro/SB16)
PNPA01B	Trantor 128 SCSI controller
PNPA01D	Trantor T160 SCSI controller
PNPA01E	Trantor T338 Parallel SCSI controller
PNPA01F	Trantor T348 Parallel SCSI controller
PNPA020	Trantor Media Vision SCSI controller
PNPA022	Always IN-2000 SCSI controller
PNPA02B	Sony proprietary CD-ROM controller
PNPA02D	Trantor T13b 8-bit SCSI controller
PNPA02F	Trantor T358 Parallel SCSI controller
PNPA030	Mitsumi LU-005 Single Speed CD-ROM controller + drive
PNPA031	Mitsumi FX-001 Single Speed CD-ROM controller + drive
PNPA032	Mitsumi FX-001 Double Speed CD-ROM controller + drive

Sound, Video Capture, and Multimedia

Device ID	Description
PNPB000	Sound Blaster 1.5-compatible sound device
PNPB001	Sound Blaster 2.0-compatible sound device
PNPB002	Sound Blaster Pro-compatible sound device
PNPB003	Sound Blaster 16-compatible sound device
PNPB004	Thunderboard-compatible sound device
PNPB005	Adlib-compatible frequency modulation (FM) synthesizer device
PNPB006	MPU401 compatible
PNPB007	Microsoft Windows Sound System-compatible sound device
PNPB008	Compaq Business Audio
PNPB009	Plug and Play Microsoft Windows Sound System device
PNPB00A	MediaVision Pro Audio Spectrum (Trantor SCSI-enabled, Thunder Chip-disabled)
PNPB00B	MediaVision Pro Audio 3-D

Continued

Sound, Video Capture, and Multimedia *(continued)*

Device ID	Description
PNPB00C	MusicQuest MQX-32M
PNPB00D	MediaVision Pro Audio Spectrum Basic (no Trantor SCSI, Thunder Chip-enabled)
PNPB00E	MediaVision Pro Audio Spectrum (Trantor SCSI-enabled, Thunder Chip-enabled)
PNPB00F	MediaVision Jazz-16 chip set (OEM versions)
PNPB010	Auravision VxP500 chip set–Orchid Videola
PNPB018	MediaVision Pro Audio Spectrum 8-bit
PNPB019	MediaVision Pro Audio Spectrum Basic (no Trantor SCSI, Thunder chip-disabled)
PNPB020	Yamaha OPL3-compatible FM synthesizer device
PNPB02F	Joystick/gameport

Modems

Device ID	Description
PNPC000	Compaq 14400 modem (TBD)
PNPC001	Compaq 2400/9600 modem (TBD)

A P P E N D I X C

Accessibility

This appendix presents recommendations for computer and component design related to lowering access barriers to computer use for persons with special physical needs.

These guidelines were developed in consultation with the Trace Research and Development Center at the University of Wisconsin-Madison and were based on research funded by the National Institute for Disability and Rehabilitation Research (NIDRR). For more information, see the references at the end of this appendix.

For information about software accessibility guidelines, see the web site at http://www.microsoft.com/enable/.

Contents

Introduction

Personal computers are powerful tools that enable people to work, create, and communicate in ways that might otherwise be difficult or impossible. The vision of making computers easier for everyone to use, however, can be realized only when people with disabilities have equal access.

Computer accessibility has become an increasingly important issue in the home and workplace. An estimated eight out of ten major corporations employ people with disabilities who might need to use computers as part of their jobs. In the United States alone, more than 30 million people have disabilities that affect PC accessibility. In addition, as the post-war generation ages and more people experience functional limitations, computer accessibility will become a more important issue. Addressing disabilities in design will benefit all users by simplifying tasks.

In the United States, legislation such as the Americans with Disabilities Act and Section 508 of the Rehabilitation Act has brought accessibility to national attention in both the public and private sectors. Accessibility is also being incorporated into official and international standards for usability, such as ANSI 200. Such recommendations affect the following:

- Visual displays and indicators
- Sound
- Manipulation and physical design
- Input devices and software controls
- Labeling
- Documentation

What Are Disabilities?

Individuals are not disabled; rather, some people have difficulties performing certain tasks, such as using a mouse or reading small print. When these limitations are serious enough to impact the person's performance, they are referred to as "disabilities." Anyone can experience the same difficulties because of illness, accident, environment (such as loud background noise), or hardware error (such as a missing mouse).

Disabilities can be classified into the following general categories:

- **Visual impairment.** This ranges from slightly reduced visual acuity to total blindness. Those with reduced visual acuity might only need images to be reasonably sized or specially enlarged, or they might need high contrast between foreground and background. Users with more severe impairments might require that output be translated into audible cues, spoken text, or Braille.
- **Hearing impairment.** Some individuals do not hear beeps, distinguish different sounds, or recognize spoken words. These users might need the computer to prompt them in a different manner, such as a screen flash, an indicator lamp, or an on-screen message.
- **Movement impairment.** Some users are unable to perform certain manual tasks, such as using a mouse or typing two keys at the same time. Others might have a tendency to hit multiple keys, might "bounce" fingers off keys, or might be unable to hold a printed book. Many users need keyboards and mouse functions adapted to their requirements, or they may rely exclusively on a single input device.
- **Cognitive impairment.** Cognitive impairments take many forms, including memory loss, perceptual differences, and conditions such as Downs syndrome. Language impairments such as dyslexia or illiteracy are also very common. Those who speak English as a second language can be considered to have a form of language impairment. Proper design can increase accessibility for these computer users.
- **Seizure disorders.** People with some forms of epilepsy might experience minor or severe seizures when a monitor flashes at certain rates or they hear certain types of random or repetitive sounds.
- **Speech impairments.** Although speech difficulties do not normally affect a person's ability to use a computer, it can be a problem in using telecommunications and voice menus. And in the future, speech difficulties might affect normal computer usage if voice recognition becomes a common form of input.

What Is Accessibility?

Accessibility means making computers accessible to a wider range of users than would otherwise be the case. Special needs can be addressed in several ways:

- New features built into hardware and operating systems help make computers accessible to users with and without specialized needs. These solutions, often referred to as “electronic curb cuts,” are preferred because the features are available on all workstations and can be used with all applications.
- Usability features can be built into mainstream products, making them easier to use for people with disabilities. Examples include customizable colors and keyboard accelerators. In many cases, these features also benefit people without disabilities.
- Utilities that upgrade a system make the PC more usable by people with disabilities. Examples of utilities include Braille-output systems for people who are blind or software that modifies the behavior of the keyboard and mouse.
- Specialized applications, such as a word processor designed to integrate voice and text, help individuals with limited reading and writing skills.

A variety of hardware and software products have been designed to help people with disabilities make use of PCs. Following are some of the different products available for upgrading the accessibility of the Microsoft Windows operating system:

- Programs that enlarge or alter the color of information on the screen
- Programs that describe on-screen information in Braille or synthesized speech
- Hardware and software utilities that modify the behavior of the mouse and the keyboard
- Programs that enable users to “type” using a mouse or their voice
- Word or phrase prediction software that allows quicker typing with fewer keystrokes
- Alternative input devices, such as single-switch or puff-and-sip devices

Visual Displays and Indicators

Visual display is the predominant form of display on today’s computers. This includes the standard display screen, light-emitting diode (LED) or liquid crystal display (LCD) icon displays on or near the keyboard, and special visual indicators on peripheral devices.

For people with low vision or blindness, these displays are a barrier to computer use. Special screen-magnification software can increase the image size for people with low vision. Similarly, software “screen readers” can access information and read it aloud to users who are blind. Information provided by indicator lights or LCD mini-displays, however, must also be made available and readable.

Hardware design strategies for providing greater access to visual information include the following:

1. Minimize glare.

   Glare caused by reflections or mismatched color combinations, overly bright indicators, and so on might present problems for users with low vision. Minimizing glare allows these users greater access to displays.
2. Avoid 5-Hz to 50-Hz refresh rate or flicker rate.

   This allows users with photosensitive epilepsy, who might have a seizure if exposed to strong stimuli in the 50-Hz range, to more safely use the system.
3. Make LCD and LED indicators, warning lights, and alert lights software-readable.

   This allows users who are blind to use their screen-reading software to access important indicators, warnings, and notices.

The following standard design practices also facilitate accessibility:

4. Provide contrast and brightness controls.

   This is important for users with low vision and color blindness, making it possible for them to adjust the display to accommodate their needs and preferences.
5. Provide a display connector for an external or additional monitor.

   Users with low vision often need to augment the system with a larger monitor to take full advantage of screen-magnification software. This is especially true for mobile PC users.
6. Provide adjustable monitors.

   Users with physical disabilities adjust the angle and position of the monitor to suit their seating position and approach to the system. Other users with low vision often adjust the monitor to reduce glare.

Sound

Sound is increasingly being used to convey information important to the computer operator. This includes alerts in addition to speech and other complex audio feedback. This can pose problems for any user on an airplane or in other noisy environments, and it especially can cause problems for those who are deaf or hard of hearing.

Hardware design strategies for providing greater access to aural information include the following:

1. Provide a headphone jack.

   Headphone jacks allow users with reduced hearing to block out background noise and make the output louder (by using headphones or by connecting directly to their hearing aids).

2. Maximize the range of volume adjustment.

   Volume controls allow users with reduced hearing to adjust the volume level to suit their needs.

3. Direct speakers toward the user.

   This maximizes the signal-to-noise ratio (SNR) for all users. This is especially important for users with hearing loss.

4. Provide a visually distinct indicator for all alerts or warning sounds created by hardware.

   Visual indicators make recognizing alerts easier for users working in loud environments or for users with deafness or reduced hearing. (Software-generated alerts should be handled by software.)

5. Provide a visual indication of important sounds generated in normal computer operation.

   Some natural sounds, such as those generated by a disk drive or printer, can be important to system operation. Where this is true, some visual indication of the sound should be provided so users in loud environments or those with reduced hearing or deafness can effectively use the systems.

6. Send hardware-generated beeps to the operating system.

   This allows visual display of beeps for users in loud environments, users located in another room, or users with reduced hearing or deafness. Examples include beeps related to the keyboard and printer.

7. Place the microphone in the orientation recommended by the microphone manufacturer.

   Along with speech-recognition software, properly locating the microphone makes the computer more accessible to users who are blind, have low vision, are physically disabled (including those with repetitive strain injury [RSI] and carpal tunnel syndrome), or have difficulty writing.

8. Include a speech-capable sound system.

   Such a sound system provides speech capabilities for nonspeaking persons. It also provides the necessary hardware support for a speech-based access system used by people with low vision, blindness, or reading difficulties.

9. Reduce noise level of operating internal components.

   This increases the ability of users with neural hearing loss or hearing aids to converse or use computer sound-based features.

Manipulation and Physical Design

Often, accessibility efforts focus on the input or display components of a computer. However, many of the physical design characteristics of the computer are equally important. For example, if the person is unable to operate the latches to open a portable computer, other aspects of the computer's design are of little importance.

Some users have conditions that result in weakness or poor motor control. Some have use of just one hand or reduced range of motion or reach. Hardware design strategies for providing greater access include the following:

1. Eject media a sufficient distance for grasping.

   Ejecting 0.5 to 0.75 inch or more is helpful for those with reduced hand functions.

2. Sculpt or bevel device entry slots.

   This assists in guiding the disk into the slot for those with reduced motor control. This also gives a tactile indicator, which helps those with low vision locate the slot. CDs should seat themselves properly when dropped into trays (that is, no fine positioning is required from the user). The CD should also be easy to remove from the tray (that is, slots at the sides of the tray allow for placing a finger under the disc to lift it out).

3. Make latches operable with one hand.

   This allows those who do not have use of limbs or who use assistive pointing devices to operate latches.

4. Minimize the force required for inserting and retrieving media.

   This assists those with reduced strength and grasp capabilities. A maximum force of 2 Newtons is advised, but it is preferable for the mechanism to “suck” the disk in for insertion and to not require a force of more than 2 Newtons for pulling it out of the slot.

5. Ensure that media stands up to rough handling, and use caddies for media that do not.

   This allows individuals with poor motor control to safely handle all media without inflicting irreversible damage.

6. Provide error flags for misinserted media, especially CDs.

   When it is not possible to block misinserted media (see item 8 below), the hardware could contain a disk-present detector to warn the user that a disk has been inserted but is unreadable because it is upside-down (and not just in an unreadable format). This would be an asset for users with visual impairments and as well as for novice users.

7. Ensure that devices do not generate electromagnetic or radio frequency (RF) fields that would affect users with hearing aids.

   Electromagnetic fields can couple with induction pick-ups in hearing aids, causing loud or disturbing noises. RF can affect all hearing aids. There are currently no industry standards for these levels, so reasonable care should be taken and testing is recommended.

The following design practices are fairly standard and also facilitate access and use by users with disabilities:

8. Use media misinsertion blocking.

   Blocking assists all users as a memory aid, but it especially helps those with low vision or cognitive impairments who might forget or misinterpret how to insert media.

9. Offer components that allow for use of alternative input devices.

   This assists those who use alternative input or output devices because of the nature of their reduced ability.

10. Provide adjustable height, swivel, and so on where appropriate.

    Adjustable components assist those who do not have a full range of movement; displays and input devices can be oriented toward the user.

11. Minimize operation noise levels.

    In general, quieter components (fans, disk drives, and so on) are easier to use for users with neural hearing loss or hearing aids.

12. Eliminate hard edges or sharp corners that could cause injury or inhibit correct device placement.

    Softening component edges assists those with reduced motor control, minimizing the potential for injury.

13. Manufacture outer surfaces using only hypoallergenic materials.

    Chromium and nickel are known to cause allergic reactions in some users and should therefore be avoided in any part that could come into contact with users’ skin during normal use.

Input and Controls

The ability to operate a computer directly depends on the ability to use its input devices and controls. For many people with physical or visual disabilities, using a computer depends either on the design of the input and control devices or on the computer's ability to substitute other control mechanisms (such as use alternative input devices or software controls). Users who are blind cannot use input mechanisms that require hand-eye coordination, such as a mouse or a control with no tactile or auditory reference.

Easier-to-operate controls are appreciated by all users, as is the ability to connect and use alternative preferred input devices.

Hardware design strategies for providing greater access to input and control functions include the following:

1. Allow connection, substitution, or addition of alternative input devices.

   A second serial port is helpful for those using SerialKeys software in Windows 95 to provide alternative access, as well as for blind users who typically use speech synthesizers with serial connections. For users who cannot use standard input devices (even with software extensions such as StickyKeys), the availability of an external keyboard and mouse connection on portable systems allows the user to substitute specially designed keyboards or pointing devices.
2. Design all controls to operate from the keyboard.

   This allows users with restricted reach or motor skills to operate controls on the CPU, monitor, and so on that they would otherwise be unable to use. It also allows these users to operate the controls from any other keyboard they might use.
3. Mount all controls on the front of the device.

   By placing all controls facing the user, disabled users have better access.
4. Limit button design primarily to push-button controls.

   Push-button controls assist those with reduced motor control and those using head or mouth sticks or other alternative pointing devices.
5. Use concave buttons, especially where sustained force is required.

   Concave buttons help keep fingers or pointers from slipping, assisting those with reduced finger or motor control (such as tremor) and those who must use headsticks or other pointing devices.
6. Avoid twisting motions.

   Users with some disabilities, such as cerebral palsy or arthritis, find twisting motions difficult or impossible. Instead, use push-button or edge controls.

7. Minimize force required for operation.

   A maximum force of 2 Newtons for any operation is advised. For controls, use a light touch or substitute manual mechanisms with power-driven mechanisms to achieve sustained or heavy touch.

8. Make all controls operable with one hand.

   This allows individuals who have only one hand or who use assistive pointing devices to operate the controls.

9. Avoid capacitance-based controls.

   Capacitance-based controls require contact with the human skin. Avoiding these types of controls allows users with assistive pointing devices or artificial limbs to use the system.

10. Use functional grouping and layout of controls.

    Functional grouping assists those with low vision or cognitive impairments to quickly find the right keys. Examples of functional sets are direction keys and control keys. The keys can be grouped by color (making sure to take color blindness into account) or by other design characteristics, such as shape or feel.

11. Make controls tactually discernible.

    This includes locator ridges or nibs on the home keys. Flat-membrane keypads with no tactile features should be avoided because they provide no feedback on the location of the buttons.

12. Make the state of non-momentary controls tactually discernible.

    This allows those who have low vision or who are blind to discern the state of a control.

13. Provide tactile and audible actuation feedback for controls.

    People who have low vision or who are blind require non-visual signals to determine when a key has been pressed.

14. Avoid keys that cannot be read or simulated by software.

    Many users rely on software programs that either detect or simulate keystrokes. For example, the StickyKeys, SlowKeys, and SerialKeys features in Windows 95 rely on this ability when compensating for a user's difficulty with the keyboard. Blind users can also query the state of toggle keys using software. Any nonstandard keys should produce scan codes that trigger their functionality.

15. Maximize size of controls within the space available.

    This makes operation easier for those with low vision and those with limited dexterity who might have difficulty manipulating small controls.

16. Space the controls a sufficient distance to allow for tactile and visual discrimination.

    This facilitates access by users with visual impairments as well as assisting those with reduced motor control. For example, on keyboards, key-top spacing should be approximately one-half of the key width. For small controls, spacing can be tighter, but functional grouping should be maintained to minimize pressing several controls at once.

17. Design stable controls.

    Unintended activation should not change the adjustment state of the control. Some users with reduced motor control or blindness might inadvertently change the setting of one control while activating another. Controls or input devices should be designed to prevent this.

18. Manufacture input guards or provide mounting for guards.

    Keyguards go over the keyboard and allow users to press one key at a time while resting their hands on the keyguard surface. They allow those with reduced strength or reduced motor control to have more control over their input. Guards can also be made for other input devices, where applicable. The FilterKeys features in Windows 95 allows direct access for many users, but others benefit further from hardware guards. Manufacturers should also ensure grooves or holes are made in the edge of the device to allow for mounting of a guard.

19. Provide stable keyboards.

    Features such as non-slip feet assist those with reduced motor control who might otherwise inadvertently move the keyboard when trying to use it.

20. Remove left-right bias.

    Designing equipment with no left-right bias assists those who have use of only one hand or who are left-handed. Where this is not possible (for example, built-in numeric keypads) alternatives should be provided (for example, support for external numeric keypads).

Labeling

The ability to read labels is important to ensure that users are able to detect and recognize controls, connectors, and media interfaces. Users with low vision or blindness often have difficulty if these labels are printed in small or low-contrast type. All users have difficulty with small labels on the back or in hard-to-view areas of the product.

Hardware design strategies for providing greater access to labels include the following:

1. Use large, high-contrast, bold stroke, sans serif lettering, and avoid using artwork behind text.

   This allows users with low vision or reduced vision to more easily read the lettering.
2. Use tactually distinct icons for controls, connectors, and legends.

   Raised icons (raised at least 1/32 inch) allow users who are blind to more easily discern an item's label by touch. They also allow any user to identify a component located out of view (such as on the back of the unit).
3. Provide optional Braille and tactile labels.

   This allows users who are blind the option of customizing labels on the unit.

Documentation

Modern multimedia computers and peripherals require more documentation than ever. For users with low vision or blindness, standard print manuals might be unusable. Users with physical disabilities might be unable to hold perfect-bound manuals open. All users would welcome manuals that are easier to understand.

Documentation design strategies for providing greater access and usability include the following:

1. Provide electronic documentation (ASCII).

   This enables generation of Braille, speech, and variable-sized text outputs for users with blindness, low vision, and cognitive or physical disabilities.
2. Provide text descriptions of graphical information.

   Written descriptions of illustrations, graphs, and so on allow users with blindness or low vision to access the information in the graphic.
3. Use clear, simple language.

   Clear and concise writing benefits everyone, but it is especially helpful to those for whom English is a second language (including sign language) or for users with cognitive disabilities.

4. Bind manuals in a way that allows the manual to lie open.

   A manual that lies flat is easier to manipulate by the user who has one hand or uses a mouthstick.

5. Provide manuals in alternative formats such as large print or Braille.

   This allows users with blindness or low vision access to the information. Recommended large print is 18-point sans serif.

6. Use high-contrast layouts.

   Users with color blindness or low vision require high contrast to access printed information.

7. Use colors that reproduce well on copy machines.

   Avoid colors that copy all gray or low contrast. Users with low vision often enlarge the information in manuals using a copy machine.

8. Provide online help.

   Online help allows users to access information without having to refer to manuals.

9. Avoid conveying information by color alone.

   Users with low vision or color blindness may have difficulty perceiving certain colors. Users who are blind scan the documentation to convert it to ASCII text.

Accessibility Recommendations for PC Design

This section presents summary lists of recommendations for design issues related to access for persons with disabilities. Item numbers are based on lists in related sections of this appendix.

1. Accessibility recommendations for physical design (casing)
Recommended

Manipulation and Physical Design:

2 Sculpt or bevel device entry slots.

3 Make latches operable with one hand.

7 Ensure devices do not generate electromagnetic or RF fields that would affect users with hearing aids.

9 Offer separate components that allow for use of alternative devices.

10 Provide adjustable height, swivel, and so on where appropriate.

11 Minimize operation noise levels.

12 Eliminate hard edges or sharp corners that could cause injury or inhibit correct device placement.

Input and Controls:

4 Limit button design primarily to push-button controls.

5 Use concave buttons, especially where sustained force is required.

6 Avoid twisting motions.

7 Minimize force required for operation.

8 Make all controls operable with one hand.

11 Make controls tactually discernible.

13 Provide tactile and audible actuation feedback for controls.

15 Maximize size of controls within the space available

16 Space the controls a sufficient distance to allow for tactile and visual discrimination.

Labeling:

1 Use large, high-contrast, bold stroke, sans-serif lettering, and avoid using artwork behind text.

2 Use tactually distinct icons for controls, connectors, and legends.

3 Provide optional Braille and tactile labels.

Documentation:

All points in the Documentation section of this appendix.

Accessibility for PC Card

This section presents summary lists of recommendations for design issues related to access for persons with disabilities. Item numbers are based on lists in related sections of this appendix.

2. Accessibility recommendations for PC Cards
Recommended

Labeling:

1 Use large, high-contrast, bold stroke, sans-serif lettering, and avoid using artwork behind text.

2 Use tactually distinct icons for controls, connectors, and legends.

3 Provide optional Braille and tactile labels.

Documentation:

All points in the Documentation section of this appendix.

Accessibility Guidelines for Input Components

This section presents summary lists of recommendations for design issues related to access for persons with disabilities. Item numbers are based on lists in related sections of this appendix.

3. Accessibility recommendations for pointing devices

Recommended

Manipulation and Physical Design:

9 Offer separate components that allow for use of alternative input devices.

12 Eliminate hard edges or sharp corners that could cause injury or inhibit correct device placement.

13 Manufacture outer surfaces using only hypoallergenic materials.

Input and Controls:

1 Allow connection, substitution, or addition of alternative input devices.

2 Design all controls to operate from the keyboard.

4 Limit button design primarily to push-button controls.

5 Use concave buttons, especially where sustained force is required.

6 Avoid twisting motions.

7 Minimize force required for operation.

8 Make all controls operable with one hand.

9 Avoid capacitance-based controls.

10 Use functional grouping and layout of controls.

11 Make controls tactually discernible.

13 Provide tactile and audible actuation feedback for controls.

16 Space the controls a sufficient distance to allow for tactile and visual discrimination.

17 Design stable controls.

18 Manufacture input guards or provide mounting for guards.

20 Remove left-right bias.

4. Accessibility recommendations for keyboards

Recommended

Visual Displays and Indicators:

3 Make LCD and LED indicators, warning, and alert lights software-readable**.**

Manipulation and Physical Design:

9 Offer separate components that allow for use of alternative input devices.

10 Provide adjustable height, swivel, and so on where appropriate.

12 Eliminate hard edges or sharp corners that could cause injury or inhibit correct device placement.

13 Manufacture outer surfaces using only hypoallergenic materials.

Input and Controls:

1 Allow connection, substitution, or addition of alternative input devices.

2 Design all controls to operate from the keyboard.

3 Mount all controls on the front of the device.

4 Limit button design primarily to push-button controls.

5 Use concave buttons, especially where sustained force is required.

6 Avoid twisting motions.

7 Minimize the force required for operation.

8 Make all controls operable with one hand.

9 Avoid capacitance-based controls.

10 Use functional grouping and layout of controls.

11 Make controls tactually discernible.

12 Make the state of non-momentary controls tactually discernible.

13 Provide tactile and audible actuation feedback for controls.

14 Avoid keys that cannot be read or simulated by software.

15 Maximize size of controls within the space available.

16 Space the controls a sufficient distance to allow for tactile and visual discrimination.

18 Manufacture input guards or provide mounting for guards.

19 Provide stable keyboards.

Labeling:

1 Use large, high-contrast, bold stroke, sans serif lettering, and avoid using artwork behind text.

2 Use tactually distinct icons for controls, connectors, and legends.

3 Provide optional Braille and tactile labels.

Documentation:

All points in the Documentation section of this appendix.

Accessibility Guidelines for Display Monitors

This section presents summary lists of recommendations for design issues related to access for persons with disabilities. Item numbers are based on lists in related sections of this appendix.

5. Accessibility guidelines for display monitors
Recommended

Visual Displays and Indicators:

1 Minimize glare.

2 Avoid 5-Hz to 50-Hz refresh rate or flicker rate.

3 Make LCD and LED indicators, warning, and alert lights software-readable.

4 Provide contrast and brightness controls.

5 Provide a display connector for an external or additional monitor.

6 Provide adjustable monitors.

Manipulation and Physical Design:

3 Make latches operable with one hand.

7 Ensure that devices do not generate electromagnetic or RF fields that would affect users with hearing aids.

9 Offer separate components that allow for use of alternate input devices.

10 Provide adjustable height, swivel, and so on, where appropriate.

11 Minimize operation noise levels.

12 Eliminate hard edges or sharp corners that could cause injury or inhibit correct device placement.

13 Manufacture outer surfaces using only hypoallergenic materials.

Input and Controls:

2 Design all controls to operate from the keyboard.

3 Mount all controls on the front of the device.

4 Limit button design primarily to push-button controls.

5 Use concave buttons, especially where sustained force is required.

6 Avoid twisting motions.

7 Minimize the force required for operation.

8 Make all controls operable with one hand.

9 Avoid capacitance-based controls.

10 Use functional grouping and layout of controls.

11 Make controls tactually discernible.

12 Make the state of non-momentary controls tactually discernible.

13 Provide tactile and audible actuation feedback for controls.

15 Maximize size of controls within the space available.

16 Space the controls a sufficient distance to allow for tactile and visual discrimination.

17 Design stable controls.

Labeling:

1 Use large, high-contrast, bold stroke, sans serif lettering, and avoid using artwork behind text.

2 Use tactually distinct icons for controls, connectors, and legends.

3 Provide optional Braille and tactile labels.

Documentation:

All points in the Documentation section of this appendix.

Accessibility Guidelines for Audio Components

This section presents summary lists of recommendations for design issues related to access for persons with disabilities. Item numbers are based on lists in related sections of this appendix.

6. Accessibility features for headphones

Recommended

Sound:

1 Provide a headphone jack.

2 Maximize the range of volume adjustment.

5 Provide a visual indication of important sounds generated in normal computer operation.

6 Send hardware-generated beeps to the operating system.

7. Accessibility features for microphones

Recommended

Sound:

7 Place the microphone in an orientation recommended by the microphone manufacturer.

Labeling:

1 Use large, high-contrast, bold stroke, sans serif lettering, and avoid using artwork behind text.

2 Use tactually distinct icons for controls, connectors, and legends.

3 Provide optional Braille and tactile labels.

8. Accessibility features for speakers

Recommended

Sound:

2 Maximize the range of volume adjustment.

3 Direct speakers toward the user.

8 Include a speech-capable sound system.

9 Reduce noise level of operating internal components.

Accessibility Guidelines for Storage Devices

This section presents summary lists of recommendations for design issues related to access for persons with disabilities. Item numbers are based on lists in related sections of this appendix.

9. Recommendations for 3.5-inch floppy disk drives

Recommended

Visual Displays and Indicators:

3 Make LCD and LED indicators, warning, and alert lights software-readable.

Sound:

4 Provide a visually distinct indicator for all alerts or warning sounds created by hardware.

5 Provide visual indication of important sounds generated in normal computer operation.

Manipulation and Physical Design:

1 Eject media a sufficient distance for grasping.

2 Sculpt or bevel device entry slots.

3 Make latches operable with one hand.

4 Minimize the force required for inserting and retrieving media.

5 Ensure that media takes rough handling, and use caddies for media that do not.

6 Provide error flags for misinserted media, especially CDs.

7 Ensure that devices do not generate electromagnetic or RF fields that would affect users with hearing aids.

8 Use media misinsertion blocking.

11 Minimize operation noise levels.

Input and Controls:

1 Allow connection, substitution, or addition of alternative input devices.

2 Design controls to operate from the keyboard.

3 Mount all controls mounted on the front of the device.

4 Limit button design primarily to push-button controls.

5 Use concave buttons, especially where sustained force is required.

6 Avoid twisting motions.

7 Minimize the force required for operation.

8 Make all controls operable with one hand.

11 Make controls tactually discernible.

Labeling:

2 Use tactually distinct icons for controls, connectors, and legends.

3 Provide optional Braille and tactile labels.

Documentation:

All points in the Documentation section of this appendix.

10. Accessibility recommendations for CD-ROM drives

Recommended

Visual Displays and Indicators:

3 Make LCD and LED indicators, warning, and alert lights software-readable.

Sound:

4 Provide a visually distinct indicator for all alerts or warning sounds created by hardware.

5 Provide visual indication of important sounds generated in normal computer operation.

Manipulation and Physical Design:

1 Eject media a sufficient distance for grasping.

2 Sculpt or bevel device entry slots.

3 Make latches operable with one hand.

4 Minimize the force required for inserting and retrieving media.

5 Ensure that media takes rough handling, and use caddies for media that do not.

6 Provide error flags for misinserted media, especially CDs.

7 Ensure that devices do not generate electromagnetic or RF fields that would affect users with hearing aids.

8 Use media misinsertion blocking.

11 Minimize operation noise levels.

12 Eliminate hard edges or sharp corners that could cause injury or inhibit correct device placement.

Input and Controls:

2 Design all controls to operate from the keyboard.

3 Mount all controls on the front of the device.

4 Limit button design primarily to push-button controls.

5 Use concave buttons, especially where sustained force is required.

6 Avoid twisting motions.

7 Minimize the force required for operation.

8 Make all controls operable with one hand.

11 Make controls tactually discernible.

13 Provide tactile and audible actuation feedback for controls.

15 Maximize size of controls within the space available.

Labeling:

2 Use tactually distinct icons for controls, connectors, and legends.

3 Provide optional Braille and tactile labels.

Documentation:

All points in the Documentation section of this appendix.

Accessibility Guidelines for Printers

This section presents summary lists of recommendations for design issues related to access for persons with disabilities. Item numbers are based on lists in related sections of this appendix.

11. Accessibility for printers

Recommended

Visual Displays and Indicators:

3 Make LCD and LED indicators, warning, and alert lights software-readable.

Sound:

4 Provide a visually distinct indicator for all alerts or warning sounds created by hardware.

5 Provide visual indication of important sounds generated in normal computer operation.

6 Send hardware-generated beeps to the operating system.

9 Reduce noise level of operating internal components.

Manipulation and Physical Design:

1 Eject media a sufficient distance for grasping.

3 Make latches operable with one hand.

4 Minimize force required for inserting and retrieving media.

6 Provide error flags for misinserted media, especially CDs.

7 Ensure that devices do not generate electromagnetic or RF fields that would affect users with hearing aids.

11 Minimize operation noise levels.

12 Eliminate hard edges or sharp corners that could cause injury or inhibit correct device placement.

13 Manufacture outer surfaces using only hypoallergenic materials.

Input and Controls:

1 Allow connection, substitution, or addition of alternative input devices.

2 Design all controls to operate from the keyboard.

3 Mount all controls on the front of the device.

4 Limit button design primarily to push-button controls.

5 Use concave buttons, especially where sustained force is required.

6 Avoid twisting motions.

7 Minimize the force required for operation.

8 Make all controls operable with one hand.

9 Avoid capacitance-based controls.

10 Use functional grouping and layout of controls.

11 Make controls tactually discernible.

12 Make the state of non-momentary controls tactually discernible.

13 Provide tactile and audible actuation feedback for controls.

15 Maximize size of controls within the space available.

16 Space the controls a sufficient distance to allow for tactile and visual discrimination.

20 Remove left-right bias.

Labeling:

1. Use large, high-contrast, bold stroke, sans serif lettering, and avoid using artwork behind text.
2. Use tactually distinct icons for controls, connectors, and legends.
3. Provide optional Braille and tactile labels.

Documentation:

All points in the Documentation section of this appendix.

Accessibility References and Resources

The following represents some of the references, services, and tools available to help build hardware and software that addresses accessibility needs.

Accessibility Publications

The following publications provide supplementary information.

Berliss, J. R. *Checklists for implementing accessibility in computer laboratories at colleges and universities.* University of Wisconsin-Madison, Trace Research and Development Center, 1990.

Fontaine, P. *Writing Accessible HTML Documents.* General Services Administration, 1995.

Lowney, G. C. *The Microsoft Windows Guidelines for Accessible Software Design.* Microsoft Corporation, 1995.

Novak, M. (Ed.) *General Input Device Emulating Interface: Version 1.* University of Wisconsin-Madison, Trace Research and Development Center, 1991.

Thoren, Clas (Ed.) *Nordic guidelines for computer accessibility.* Gotab, Sthlm, 1993.

Vanderheiden, G. C. *Consideration in the design of computers and operating systems to increase their accessibility to persons with disabilities.* University of Wisconsin-Madison, Trace Research and Development Center, 1988.

———. *Making software more accessible to people with disabilities.* University of Wisconsin-Madison, Trace Research and Development Center, 1991.

——— and Vanderheiden, K. R. *Accessibility Design Guide I: Guidelines for the Design of Consumer Products to Increase Their Accessibility to Persons with Disabilities or Who Are Aging.* University of Wisconsin-Madison, Trace Research and Development Center, 1991.

Resources for Accessibility Design

This section lists some documents and services related to accessibility design.

Microsoft Windows Guidelines for Accessible Software Design
Describes techniques for developing software applications that are usable by people with disabilities. This document is included on the Microsoft Developer Network (MSDN) CD-ROM. To obtain additional copies or for information about other Microsoft products and services for people with disabilities, contact:

Microsoft Sales Information Center
One Microsoft Way
Redmond, WA 98052-6393
Voice: (800) 426-9400
Text Telephone: (800) 892-5234
Fax: (206) 635-6100

Research and Product Information
For information on research and development concerning technology, communication, and disabilities, or for catalogs of accessibility products and service providers, contact:

Trace Research and Development Center
Waisman Center and Department of Industrial Engineering
University of Wisconsin
Madison, WI 53705
E-mail: info@Trace.Wisc.Edu
Fax: (608) 262-8848
FTP, Gopher, and WWW servers: trace.wisc.edu

For a list of listserve discussions, send “LISTS” to listproc@trace.wisc.edu.

Documentation in Accessible Formats
Recording for the Blind and Dyslexic, Inc., is an organization that can assist in preparing documentation in accessible formats, including electronic text and audio tape. Contact:

Recording for the Blind and Dyslexic, Inc.
20 Roszel Road
Princeton, NJ 08540
Voice telephone: (800) 221-4792
Fax: (609) 987-8116

For a list of organizations that can assist in converting documentation into large print or Braille, contact the Microsoft Sales Information Center at the address listed above.

Assistive Technology Programs
For general information or recommendations for how computers can help specific users, consult a trained evaluator. An assistive technology program will provide referrals to programs and services that are available to you. To locate the assistive technology program nearest you, contact:

National Information System
Center for Developmental Disabilities
Midland Center
Education Building
8301 Sarrow Road
Columbia, SC 29203
Voice or text telephone: (803) 777-4435
Fax: (803) 935-5250

Closed Captioning and Video Description
The following service providers can assist in adding closed captions or video description to video tape and film:

The Caption Center
125 Western Avenue
Boston, MA 02134
Voice/text telephone: (617) 492-9225
Fax: (617) 562-0590

National Captioning Institute
5203 Leesburg Pike, Suite 1500
Falls Church, VA 22041

APPENDIX D

Legacy Support

This appendix summarizes assignments for interrupt request (IRQ), direct memory access (DMA), and I/O port addresses used by built-in devices on legacy system boards. This appendix also includes requirements for any Industry Standard Architecture (ISA) legacy implementations.

Contents

Fixed ISA Interrupts

The following IRQs are used by ISA devices and are considered to be fixed assignments.

Fixed ISA Interrupts

Hardware IRQ	Default assignment
IRQ 0	System timer
IRQ 1	Keyboard
IRQ 2	Second programmable interrupt controller (PIC) cascade
IRQ 3	COM 2
IRQ 4	COM 1
IRQ 5	Sometimes LPT 2—not considered fixed
IRQ 6	Standard floppy disk controller (FDC)
IRQ 7	LPT 1
IRQ 8	Real-time clock/CMOS
IRQ 9	—
IRQ 10	Sometimes COM 4—not considered fixed
IRQ 11	Sometimes COM 3—not considered fixed
IRQ 12	PC/2-style mouse
IRQ 13	Coprocessor
IRQ 14	Primary Integrated Device Electronics (IDE) controller
IRQ 15	Secondary IDE controller

Legacy ISA DMA Assignments

The following table lists DMA channel assignments that are used by legacy ISA devices and are therefore considered fixed.

Legacy ISA DMA Considered Fixed

Hardware DMA	System function (default)
DMA 0	ISA expansion
DMA 1	—
DMA 2	FDC
DMA 3	extended capabilities port (ECP) parallel port on LPT 1
DMA 4	DMA controller cascading
DMA 5	—
DMA 6	—
DMA 7	—

Legacy ISA I/O Address Assignments

The following table lists I/O addresses that are used by legacy ISA devices and are therefore considered fixed.

Legacy ISA System I/O

I/O Address	Default system function
0000–000F	Slave DMA
0010–0018	System
0001F	System
0020–0021	Master 8259
0040–0043, 0048–004B	Programmable interrupt timer (PIT) #1, PIT #2
0050–0052	System
0060	Keyboard/mouse controller
0061	System control port B
0064	Keyboard/mouse status
0070–0071	Nonmaskable Interrupt (NMI) enable/real-time clock
0081–008B	DMA page registers
0090–0091	System
0092	System control port A
0093–009F	System

Continued

Legacy ISA System I/O *(continued)*

I/O Address	Default system function
00A0–00A1	Slave interrupt controller
00C0–00DE	Master DMA controller
00F0–00F1	Coprocessor busy clear/reset
0170–0177	Secondary IDE controller
01F0–01F7	Primary IDE controller
0201	Joystick interface
0220–022F	Sound Blaster
0278–027A	LPT 2 (XT parallel port 3)
02E8–02EF	Alternate COM (4)
02F8–02FF	COM 2
0330–0331	MPU-401
0376	IDE Controller
0378–037A	LPT 1 (XT parallel port 2)
0388–038B	Frequency modulation (FM) synthesis
03B0–03BB	MDA, EGA/video graphics array (VGA)
03BC–03BE	LPT 3 (XT parallel port 1)
03C0–03DF	EGA/VGA
03E0–03E7	PCIC PCMCIA controllers
03E8–03EF	Alternate COM (3)
03F0–03F7	FDC
03F8–03FF	COM 1
0534–0537	Windows Sound System-compatible
0CF8–0CFB	Peripheral Component Interconnect (PCI) ports

Plug and Play ISA System Requirements

Although in general ISA devices are excluded from PC 98, many PC 98 systems will include ISA support that allows users to insert ISA add-on devices. This section summarizes the basic requirements for a PC system that includes the ISA bus. In PC 97, ISA was required to be implemented with full support for Plug and Play ISA boot devices.

In addition to ISA expansion cards, the following are also ISA devices:

- 8042 and similar controllers, ports, keyboards, and mice
- DMA controllers and slaves
- FDCs
- Interrupt controllers
- Legacy parallel and serial ports
- Math coprocessors
- PITs
- VGA controllers

Any such devices located at I/O addresses below 100h can use fixed resources and are exempt from Plug and Play requirements for unique IDs, flexible resource configuration, and dynamic disable capabilities.

1. System supports Plug and Play ISA specification and Plug and Play BIOS
Required
If ISA support is included in a PC 98 system, the manufacturer must implement the standards described in the following Plug and Play specifications:

- *Plug and Play ISA Specification, Version 1.0a*
- *Plug and Play BIOS Specification, Version 1.0a*
- *Clarifications to the Plug and Play BIOS Specification, Version 1.0a.*

The Plug and Play specifications are available at http://www.microsoft.com/hwdev/specs/pnpspecs.htm. Additional ISA clarifications and white papers related to ISA Plug and Play under the Microsoft Windows operating system are available at http://www.microsoft.com/hwdev/busbios/.

Note: Standard system devices are excluded from this requirement. The system can reserve static resources for devices such as interrupt controllers 1 and 2, timer (8254-2), keyboard controller (8042), real-time clock, DMA page registers, DMA controllers 1 and 2, and math coprocessor (if present). For a system based on Intel Architecture, these fixed resources are located at I/O addresses below 100H and can also include an NMI mask.

Plug and Play ISA Device Requirements

This section includes additional requirements for ISA cards, including requirements for design implementations that appear only as recommendations in the ISA specification, to ensure that such cards will perform correctly under Windows.

The information in this section is provided for manufacturers of ISA devices who want to ensure that their devices are completely compatible with Plug and Play operating systems.

For more details, see the Plug and Play ISA specification.

2. ISA device complies with Plug and Play ISA standards

Required

Any card or bus that implements Plug and Play ISA must fully implement the standards defined in the *Plug and Play ISA Specification, Version 1.0a.* This specification also defines the requirements for a unique ID for each ISA device. The unique ID is used to identify the device for Plug and Play configuration.

3. Option ROMs are used only on cards with boot devices

Required

This requirement applies only for x86-based systems. Option ROMs must be used only on cards that contain boot devices.

Cards with option ROMs must not hook the primary boot interrupts (Int 9h, Int 10h, Int 13h, Int 18h, and Int 19h) until the system calls the boot connection vector in the selected option ROM expansion header.

For cards with option ROMs, the default configuration must be able to be disabled after the card has been isolated.

4. Implement full 16-bit I/O address decode logic

Required

This circuit can be simple enough to limit I/O addresses to the 0h to 3FFh range, or it can be flexible enough to use the upper address regions. For more information, see the "Basic PC 98" chapter in Part 2 of this guide.

5. ISA device and driver support IRQ sharing if resource requirements cannot be met

Required

This requirement does not apply for Windows NT drivers. This is a requirement only if the device cannot meet the PC 97 resource requirements (as defined for the particular device class in the related chapter in Part 4 of *PC 97 Hardware Design Guide*). This requirement applies only for devices of the same class, not across device classes.

To share IRQs, the following requirements must be met:

- The IRQ line must be pulled high by the system board.
- The IRQ line must never be driven high by the devices.
- To signal an interrupt, devices must pull the IRQ line low for a minimum of 100 nanoseconds and then release it. The interrupt is signaled by the rising edge that occurs as a result of the pull-up on the IRQ line.
- The drivers for all devices connected to the IRQ line must correctly support the interrupt-sharing services of the virtual programmable interrupt controller device (VpicD). This means that after dispatching an interrupt from VpicD, the drivers must respond to VpicD and correctly indicate whether they actually processed an interrupt for their devices. VpicD will ensure that all devices with pending interrupts have been serviced before returning from the interrupt.
- IRQ sharing support implemented in the device driver for servicing interrupts.

6. Unimplemented registers return a deterministic value when read

Required

Any unimplemented registers in the range 00h–2Fh must return a deterministic value when they are read. Unimplemented configuration registers must return the “disabled” or “unused” value (not necessarily 0) when they are read.

7. Each ISA card provides complete and correct identifiers

Required

In the Plug and Play ISA specification, it is required that a Plug and Play card have both an industry-unique Vendor ID (acquired by sending e-mail to pnpid@microsoft.com) and a company-unique Product ID (assigned by the manufacturer). The specification requires that this Product ID be unique among all Plug and Play ISA cards manufactured by that company. This means each product (for example, fax card, display adapter, sound adapter, and so on) and every model (for example, 14.4 fax, 28.8 fax, and so on) from the same manufacturer must have different product identifiers.

This is a requirement because it allows the operating system to isolate and identify these different cards. The user must never have a Plug and Play card that cannot be identified because it cannot be distinguished from other models of cards from the same manufacturer. The use of a unique Product ID does not solve the problems that occur when a user installs two of the same cards in a PC system.

In those cases, the user might install a Plug and Play card but will not receive indication that it was installed and the card will not work. For this purpose, the Plug and Play ISA specification defines a unique serial-number field that can be added to the Vendor and Product IDs to make the card completely unique. A board-unique number in the serial-number field is required for ISA devices included on a system.

8. ISA system board devices are reported through the BIOS or use unique Serial ID

Required

A peripheral ISA device implemented on the system board can use a fixed Serial ID (which is not unique) if the device is reported through the BIOS.

If the system board device participates in the Plug and Play ISA isolation scheme (rather than being reported through the BIOS), then it must meet the same requirements for a unique Serial ID as for an add-on card.

Notice that it is possible that an add-on card containing an ISA chip might be added to a PC system that contains the same chip on the system board. In such a case, the add-on device will be found only if it has a different Serial ID.

9. IDs using PNP suffix are allowed only in the Compatible Device ID field

Required

Device IDs that use the three-character PNP suffix are allowed only in the Compatible Device ID field and cannot be used as Device ID or Logical Device ID fields. The exception would be the device to which the PNP-based ID was originally assigned.

Resource data describe what resources must be available for each logical device on the card (for example, number of available IRQ numbers, address ranges of memory, and so on). Resource data can be stored in the same nonvolatile storage device (such as a serial ROM) that contains the serial identifier. The resource data in the nonvolatile storage device must be sequentially loaded into the resource data register (04h).

The content of the nonvolatile storage device must be programmed with the information the system needs to interpret which resources the card requires. The structure of the data contained in the storage device is variable, depending on what resources are needed.

The resource data for a Plug and Play ISA card can be read while the card is in the Config state. This card can enter the Config state either after it has been isolated during the isolation sequence or whenever it receives a Wake (Card Select Number [CSN]) software command in which the CSN matches the CSN assigned to the card. Only one card at a time can be in the Config state.

10. Option ROMs correctly support boot devices
Required

Plug and Play ISA expansion cards that contain boot devices require some special considerations to properly boot the system. For PC 97, the system must implement support for Plug and Play ISA boot devices and option ROMs as described in the Plug and Play BIOS specification.

The types of devices required for the boot process include the primary input device (usually a keyboard), the primary output device (usually a display adapter and monitor), and any Initial Program Load (IPL) devices.

Any Plug and Play ISA expansion card that provides a boot function must be active when the system powers up. This gives non-Plug and Play systems the means for using Plug and Play ISA devices during a legacy boot process. In this case, a non-Plug and Play system BIOS will not perform the isolation sequence but will instead perform a ROM scan to detect the presence of a boot device. After the ROM scan detects the presence of an option ROM on the boot device, the system ROM will jump to the option ROM to initialize the device. The Plug and Play option ROM on the card will detect that the system BIOS is not Plug and Play-compatible and will respond accordingly. Although an initial set of static resources must be provided during this legacy boot, the Plug and Play ISA card must be capable of changing these resources using the standard Plug and Play ISA isolation and configuration process.

As required in the Plug and Play ISA specification, resource usage of a card is always reflected in the card’s configuration registers. This information allows Windows 95 to easily determine the default settings of a Plug and Play boot device. The default settings can then be overridden by the operating system with full cooperation of the device driver.

A P P E N D I X E

Network PC System Design Guidelines, Version 1.0b

E

This appendix contains the text of the Net PC specification, co-authored by Compaq Computer Corporation, Dell Computer Corporation, Hewlett Packard Company, Intel Corporation, and Microsoft Corporation.

 For the complete text of the current version of *Network PC System Design Guidelines*, see the web site at http://www.microsoft.com/hwdev/.

Contents

Introduction to Network PC

This guide presents information for engineers who build or plan to build personal computers and expansion cards under the Network PC Design Initiative.

The Network PC is a new addition to the PC family—not a replacement—using Intel architecture and other microprocessor architectures that run the Microsoft® Windows® or Windows NT® Workstation operating systems. The Net PC will reduce the cost of business computing by optimizing the design for users who do not require the flexibility and expandability of the traditional PC, and by allowing organizations to centrally manage their information technology. Although the types of business users will vary, the Net PC will be ideally suited for those involved in activities such as data entry, transaction processing, and intranet and Internet access.

Following these design guidelines for the Net PC will allow PC manufacturers to deliver products with a baseline level of manageability and interoperability, and will offer greater certainty to information technology (IT) managers that specific steps have been taken to reduce total cost of ownership (TCO).

The Net PC supports either Windows 95 or Windows NT Workstation; however, more significant TCO reductions will be realized with Windows NT Workstation 4.0 and later versions. An important benefit of the Net PC design is its assurance of a seamless migration path to the rich system-manageability capabilities that are part of Windows NT Workstation 5.0.

The Net PC is designed to be a highly manageable platform, with instrumentation, network boot capabilities, controlled and managed upgrade capabilities, and a "sealed case" that prevents end-user access for changing the system hardware or software configuration. However, the Net PC preserves the corporate investment in existing Windows-based and Windows-compatible in-house application software while extending the computing platform to support Internet and intranet software based on Java and Microsoft ActiveX™ solutions.

The Net PC defined in these guidelines provides a complete hardware, software, and operating system solution to address the PC manageability issues in corporate environments, where the benefits of PC-based computing can be preserved and enhanced through greater centralized control.

Required vs. Recommended Features

In this guide, hardware features are described as follows:

- **Required:** These are the basic hardware features that must be implemented.
- **Recommended:** These features support or improve manageability or add functionality supported by the operating systems or software layers below the operating system, such as the BIOS. Some recommendations apply to the general system, such as recommendations for improved industrial design.
- **Optional:** These features are neither required nor recommended.

In this guide, these terms have the following meanings:

- **Must** = Required
- **Should** = Recommended

Note: The requirements defined in this guide provide guidelines for designing Net PC systems. These design guidelines are not the basic system requirements for running the Microsoft Windows or Windows NT operating system.

Conventions and Terms Used in This Guide

The following conventional terms, abbreviations, and acronyms are used throughout this guide.

Add-on devices
Devices that are traditionally added to the base PC system to increase functionality, such as audio, networking, graphics, SCSI controller, and so on. Add-on devices fall into two categories: devices built onto the system board and devices on expansion cards added to the system through a system board connector such as ISA or PCI.

Desktop Management Interface (DMI)
A framework created by the Desktop Management Task Force (DMTF). DMTF specifications define industry-standard interfaces for instrumentation providers and management applications.

End user
The person who is using the Net PC to perform his or her job function, such as inputting data or running applications.

Instrumentation

A mechanism for reporting information about the state of PC hardware and software to enable management applications to ascertain and change the state of a PC and to be notified of state changes.

Intel architecture

Refers to computers based on 32-bit microprocessors that use the Pentium instruction set, such as Intel® Pentium®, Intel Pentium with MMX™ technology, Pentium Pro microprocessors, or similar processors.

Limited end-user access

In this document, several features are defined as “not accessible to end users,” meaning that the person who is using the system to input data or run other applications does not have the ability to change the configuration, purposefully or inadvertently. Specific design guidelines might be provided in some cases, but in general, this term means that the configuration can only be changed, for example, by an administrator or service technician who has special network logon privileges, special software, or special tools. This term is equivalent to “no user-serviceable parts” in consumer electronics.

PC 97

Refers to the set of design requirements defined for the “Designed for Microsoft Windows” logo program, as specified in *PC 97 Hardware Design Guide* (Microsoft Press, 1996). References to PC 97 requirements in this guide include all changes, clarifications, and timelines for implementation of PC 97 requirements as published on http://www.microsoft.com/hwdev/pc97.htm.

RISC–based

Refers to computers based on Windows NT–compatible implementations of RISC processors, including computers with Digital Alpha 21064 (EV4) or higher processors.

System administrator

The person who administers the corporate network, servers, and clients, including configuration and management of Net PC systems.

System devices

Also system board devices. Devices on the system board, such as interrupt controllers, keyboard controller, real-time clock, direct memory access (DMA) page registers, DMA controllers, memory controllers, floppy disk controller (FDC), IDE ports, serial and parallel ports, PCI bridges, and so on. In today’s PCs, these devices are typically integrated in the supporting chip set.

Web-Based Enterprise Management (WBEM)
Technology under development by BMC Software, Inc., Cisco Systems, Inc., Compaq Computer Corporation, Intel Corporation, and Microsoft Corporation, based on standards being developed by the DMTF and the Internet Engineering Task Force (IETF), to provide a mechanism for managed components to specify the information that they can provide to management applications and to provide a mechanism that management applications can use to access the information.

Win32 Driver Model (WDM)
A driver model based on the Windows NT driver model that is designed to provide a common set of I/O services and binary-compatible device drivers for both Windows NT and future Windows operating systems for specific driver classes. These driver classes include USB and IEEE 1394 buses, audio, still-image capture, video capture, and HID-compliant devices such as USB mice, keyboards, and joysticks.

Windows
Refers to the Microsoft Windows 95 operating system, including any add-on capabilities and any later versions of the operating system.

Windows Management Instrumentation (WMI)
Extensions to WDM being developed for Windows NT 5.0 and Windows 98 to provide an operating system interface through which instrumented components can provide information and notifications.

Windows NT
Refers to the Microsoft Windows NT 4.0 operating system, including any add-on capabilities and any later versions of the operating system, unless specific design issues are defined that relate to version 5.0. In this case, the version number is specifically cited.

Wired for Management (WfM)
An initiative aimed at increasing the manageability of desktop PCs and servers and improving the management software for these systems.

Zero Administration Windows
A Microsoft initiative that focuses on improving Windows and Windows NT for maximum automation of administrative tasks with centralized control and maximum flexibility.

Network PC Hardware Requirements

This section presents the system hardware requirements for Net PC design.

General System Requirements

This section presents the basic system requirements for Net PC design.

Any requirements that are defined as specific architecture or hardware implementations are stated in this way because there are no common industry-accepted benchmark tests for system performance.

1. Minimum CPU: 133-MHz Intel Pentium processor or compatible processor with similar performance, or Windows NT–compatible RISC-based processor
Required
This minimum computational capability is required to ensure that the customer is purchasing a product optimized to run Windows-based applications.

The requirement for Windows NT–compatible RISC-based systems includes Digital Alpha 21064 (EV4) or higher processors.

2. Level 2 cache with 256K minimum, for systems with Pentium or compatible processors
Required
This minimum L2 cache is required for performance on Net PC systems that use Pentium or compatible processors. This requirement does not apply for a Net PC system with a Pentium Pro or compatible processor with a built-in L2 cache, or processors whose architecture permits equivalent performance without an L2 cache.

3. Minimum RAM: 16 MB
Required
Recommended: 32 MB

This minimum memory is required to ensure that the customer is purchasing a product optimized to run Windows-based applications.

4. Upgrade capabilities for RAM and CPU
Optional
If the capability for memory and CPU upgrade is provided, this capability must not be end-user accessible.

BIOS and Remote New System Setup

A Net PC system must be capable of remote configuration and booting, even when the operating system is not loaded. This section defines the requirements that support this capability, which specify baseline capabilities required to support remote boot and remote diagnostics.

5. Limit user access in preboot modes

Required

For a Net PC system, the operating system provides the capability for centrally enabling or disabling capabilities on the system. To ensure TCO objectives, security to protect enable/disable capabilities for hardware components must also be provided before the operating system boots. The purpose of this feature is to prevent end users from accidentally or purposefully circumventing operating-system level security as applied by an administrator.

6. System BIOS support for boot devices, for Intel architecture

Required

For network adapters, the system BIOS must comply with the requirements defined in Sections 3 and 4 (as they apply to Plug and Play devices) of the Compaq, Phoenix, Intel BIOS Boot Specification, version 1.01 or higher, which describes the requirements for Initial Program Load (IPL) devices.

If a CD-ROM device is provided as a boot device in the system, the system must support No Emulation mode in “El Torito—Bootable CD-ROM Format Specification, Version 1.0” by Compaq, Intel, and Phoenix or an equivalent method that supports the Windows NT CD-ROM installation process. For information about requirements for remote management capabilities for CD-ROM, see the “Platform Management Information Requirements” section and related attachments later in this document.

7. Support Int 13h Extensions in system BIOS and option ROMs, for Intel architecture

Required

The Int 13h Extensions ensure correct support for high-capacity drives. Support for the fixed-disk access subset of Int 13h Extensions must be provided in the system BIOS and in any option ROMs for storage devices that include BIOS support.

8. BIOS boot support for USB keyboard, if USB is the only keyboard

Required

For any system that uses the Intel architecture and has a USB keyboard as the only keyboard in the system, the system BIOS must provide boot support for USB keyboards. The specification for this support is defined in *Universal Serial Bus PC Legacy Compatibility Specification*, *Version 0.9* or higher, available from http://www.teleport.com/~usb/data/usb_le9.pdf.

9. Remote new system setup and service boot supported using DHCP and TFTP as defined in Attachment A

Required

Dynamic Host Configuration Protocol (DHCP) is an open, industry standard designed to reduce complexity of TCP/IP network administration. DHCP provides methods for dynamic configuration of computers on TCP/IP networks. DHCP is specified by IETF RFCs 1533, 1534, 1541, and 1542. Trivial File Transfer Protocol (TFTP, Revision 2) to support boot image download is implemented under IETF RFC 1350.

The required implementation for remote new system setup is defined in "Attachment A: DHCP Extensions for New System Setup" in these guidelines.

10. Preboot execution environment

Required

The execution environment provided by the client for the downloaded code described in the previous requirement must conform to the description given in "Attachment B: Preboot Execution Environment" in these guidelines.

11. Remote BIOS update and revision support

Required

Recommended: Implement a mechanism to validate that the program arrived intact after download.

BIOS ROMs must be implemented to allow them to be upgraded to a new image through OEM-provided programs using either: 1) the remote new system setup mechanism that will be downloaded and executed at boot time, or 2) normal file access and execution methods when the system is fully booted into the normal operating system environment.

If they are provided, option ROMs must also be capable of being upgraded.

Power Management Requirements

Net PC systems are power-managed systems based on solutions provided under the OnNow design initiative, so that the platform enters a low-power state when not in use. This section describes the requirements that ensure the Net PC system is power managed.

12. ACPI support meets PC 97 requirements

Required

The system board must support the *Advanced Configuration and Power Interface Specification, Revision 1.0* or higher. This requirement ensures that the system correctly supports the ACPI-based Plug and Play and power management functionality.

For complete information about requirements for ACPI support, see item #4 in the "Basic PC 97" chapter of *PC 97 Hardware Design Guide*. For information about clarifications for implementation of ACPI, see the web site at http://www.microsoft.com/hwdev/pc97.htm.

13. Hardware support for the OnNow initiative

Required

Elements of the OnNow design initiative ensure that the operating system and device drivers control the state of individual devices and the system board. For complete design information and related requirements, see item #5 in the "Basic PC 97" chapter of *PC 97 Hardware Design Guide.*

14. BIOS support for the OnNow initiative, for Intel architecture

Required

This requirement applies only to systems that use the Intel architecture. For complete design information and related requirements, see item #6 in the "Basic PC 97" chapter of *PC 97 Hardware Design Guide.*

15. Wakeup on LAN supported

Recommended until 1/1/98; Required as of 1/1/98

Until January 1, 1998, for Net PCs with Ethernet or token ring network adapters, it is recommended that the system should be capable of being awakened from a lower power state for services and management. Magic Packet capability is a possible implementation. After January 1, 1998, wakeup capabilities are required for Net PCs and must be based on matching patterns specified by the local networking software, as described in "Network Wake-up Frames" and "Network Wake-up Frame Details" in *Network Device Class Power Management Reference Specification, Version 1.0* or higher.

Pattern matching-based wakeup enables any standard Windows network TCP/IP access, such as connections to shared drives and WinSock connections, as well as focused service and management applications, to “wake up” machines from lower power states. Microsoft operating system support for wake on LAN capabilities will be compliant with this pattern matching solution. There is no guarantee of functionality with these operating system services for wake on LAN solutions that are not based on *Network Device Class Power Management Reference Specification* pattern matching.

This requirement applies specifically to Ethernet and token ring adapters. *Network Device Class Power Management Reference Specification* does not support ATM and ISDN adapters.

For additional implementation guidelines, see items #34–35 in the “Network Communications” chapter of *PC 97 Hardware Design Guide*. Implementation details are described in “Network Wake-up Frames” and “Network Wake-up Frame Details” in *Network Device Class Power Management Reference Specification, Version 1.0* or higher.

Platform Management Information Requirements

Each Net PC system must be able to provide consistent and dependable platform information for use by any management application. The management solution must ensure that the Net PC is manageable in heterogeneous networking environments and that there is a basic set of management information that is guaranteed to be available for management applications. This section defines the requirements to support these capabilities.

Overview of Platform Management Information Technologies

This section briefly describes the platform management information technologies referenced in these guidelines and gives references to full definitions and descriptions of these technologies. To provide useful guidelines for products prior to the availability of technologies under development, these guidelines include specifications for both the currently available and the forthcoming technologies for providing platform management information.

To understand the relationships of the technologies referenced in these guidelines, note that platform management information technologies generally have three key elements:

- Component Instrumentation—an interface by which information is supplied by manageable platform components.
- Management Information Provider—an interface used by applications to access platform management information.
- Management Information Schema—the logical structure of the information handled by the component instrumentation and the management information provider.

The platform management information technologies referenced in these guidelines are the following, defined in alphabetical order:

- **Common Information Model.** CIM is the management information schema for WBEM. It is an object-oriented schema that is being defined by a subcommittee of the DMTF. CIM is designed to be extended for each operating environment in which it is used; as an example of this extensibility, Windows operating systems use the CIM and have added Win32 extensions.

  For CIM specifications, see http://www.dmtf.org/work/cim.html.
- **Desktop Management Interface.** DMI is a platform management information framework created by the DMTF. DMTF specifications define industry-standard interfaces for component instrumentation and management applications.

  For specifications on DMI Management Information Interfaces, see *Desktop Management Interface Specification, Version 2.00*. For specification of DMI Component Interfaces, see *Desktop Management Interface Specification, Version 2.00*. Compliance of Net PC platforms to the DMI specifications referenced in these guidelines is measured by *DMI Compliance Guidelines, Version 1.0*.
- **CIM Object Manager.** CIMOM is a software component that interacts with the CIM schema and its extensions and in turn serves as a access point for management data providers to acquire data from the schema. Microsoft will implement CIMOM within Windows NT 5.0 and Windows 98 as well as other Win32 platforms, and will create a portable, platform-independent reference implementation. Versions of CIMOM for other operating systems may be available from other vendors. CIMOM handles both schema interactions and requests from management data providers.

An example of use of CIMOM is its capability to integrate and associate management data from different sources, as in the case where one would associate a particular desktop application with its required network services, which would in turn be associated with a particular network card and a particular path through the network to the application running on a remote server. For further explanation of this example, please see the conceptual demonstration on http://wbem.freerange.com.

- **Simple Network Management Protocol.** SNMP is used widely throughout the industry as the standard under which servers, routers, hubs, and other network-based devices are managed. Enterprise-level management applications have long used SNMP as their manageability protocol because of its stability, flexibility, and wide-spread adoption.
- **Web-Based Enterprise Management.** WBEM is a set of platform management information technologies originally proposed by BMC Software, Inc., Cisco Systems, Inc., Compaq Computer Corporation, Intel Corporation, and Microsoft Corporation based on standards being developed in a number of industry bodies, including the DMTF and the IETF. WBEM is being designed to provide uniform access for management applications to management information from a variety of sources, such as DMI, SNMP, and operating system–specific component instrumentation.

  For specifications on WBEM, see http://wbem.freerange.com.
- **Win32 Extensions Schema.** This term is another name for the Win32 extensions to the CIM schema for Windows operating systems. For specifications on the Win32 Extensions Schema, see http://www.microsoft.com/management/wbem/.
- **Windows Management Interface.** WMI is an extension to WDM and is a new component instrumentation approach for Microsoft operating systems. WMI drivers have their schema built into the driver image as a resource, which enables simple, dynamic “import” of specific driver schema data into the CIM schema.

  WDM and WMI will be available in Windows NT 5.0 and Windows 98. For specifications on WMI, see http://www.microsoft.com/management/wbem/.

In the following sections, platform management information requirements are divided into two categories:

- Component Instrumentation Requirements. Describes interfaces by which instrumentation is supplied by manageable platform components.
- Management Information Providers. Describes interfaces used by applications to access platform management information.

Component Instrumentation Requirements

To ensure a basic level of manageability, a baseline set of platform management information for each Net PC needs to be available to management applications. This section addresses how system components make this information available and which information must be made available.

16. Baseline platform management information capabilities

Required

The solution for Net PC component instrumentation for WMI-capable systems is use of the WMI extensions to WDM that use the CIM and Win32 extensions schema.

The solution for advancing manageability and deploying manageable platforms with operating systems that are not WMI-capable is the DMI 2.0-based instrumentation solution. Intel and Microsoft are working together to ensure that there will be no loss of functionality as Net PCs are upgraded to use WMI-enabled operating systems.

WMI Driver Instrumentation

This section summarizes the requirements related to implementing instrumentation for WMI-capable systems using the WMI extensions.

17. Support WMI/CIM and Win32 extensions schema objects and data

Required

For Net PCs running WMI-capable operating systems, the CIM and Win32 extensions schema classes and associations listed in “Attachment H: WMI/CIM and Win32 Extensions Instrumentation Details” must be supported through WMI. Subsequent revisions of *Network PC System Design Guidelines* will incorporate additions to this set of classes and associations.

18. Support WMI alert generation for required events

Optional

None of the classes defined in the required schema have required events. Therefore, support for WMI alert generation is not required at this time, but might become required in subsequent revisions of *Network PC System Design Guidelines*.

19. Compliant with WMI alert model for WMI alerts

Required

WMI alerts that fit within the size limitation of the alert buffer can be sent directly with the alert packet. Larger alerts are required to send a global unique ID (GUID) in the alert, which is then later queried to obtain the full alert data.

All WMI alerts must supply a WMI alert severity level as recognized by the WMI alert model.

20. WMI instrumentation interface meets device-specific requirements

Required

WMI-based component instrumentation must comply with the device-specific requirements that will be described in the Microsoft Windows Device Driver Kit (DDK) WMI supplement.

DMI Component Instrumentation

This section summarizes the requirements if a DMI-based instrumentation solution is implemented.

21. DMI standard groups instrumented and deployed

Required

The standard groups listed in “Attachment I: DMI Instrumentation Details” must be instrumented and deployed on DMI-instrumented Net PCs.

22. Components compliant with DMI Component Interface

Required

DMI-instrumented components on Net PCs must comply with the Component Interface (CI) as specified for DMI version 2.0. This interface includes compliance with the Service Provider API for Components and the Component Provider API. The required DMI instrumentation must be deployed with each DMI-instrumented platform and installed with the operating system. See the DMTF compliance guidelines regarding backward compatibility for existing instrumentation implemented using the DMI version 1.x block interface.

23. DMI event generation for DMI events in required groups

Optional

Some of the required standard groups specified in “Attachment I: MI Instrumentation Details” are associated with event generation groups. Event generation for these groups is optional.

24. Compliant with DMI event model for DMI events generated

Required

If DMI-based instrumentation generates events associated with a required group, event generation must be compliant with the event model specification defined in *Desktop Management Interface Specification, Version 2.00*.

Management Information Providers

To enable management applications to access the Net PC, each Net PC needs at least one management information provider that makes management information available to management applications using common mechanisms.

25. At least one management information provider enabled
Required

When available, WBEM-based management information providers will provide uniform access to management information from a variety of sources, including platform component instrumentation. Net PCs for which WBEM-based management information providers are available will employ those providers.

Net PCs for which WBEM-based management information providers are not available will employ DMI-based management information providers.

Note that these guidelines define the minimum management information provider functionality for a Net PC. Deployment of additional management information providers on Net PCs, such as SNMP agents, might provide significant value in certain environments.

WBEM-based Management Information Provider

Having a WBEM-based management information provider allows a Net PC to be managed by applications that can access the platform through industry-standard WBEM protocols and/or interfaces.

Microsoft will implement HMMP (a protocol for which standardization is currently being proposed to the IETF) for local and remote access to CIMOM and will also implement Common Object Model (COM) for local CIMOM access in Windows NT 5.0 and Windows 98. Highest performance local accesses are achievable using the COM interface; greatest portability is achieved using HMMP.

Furthermore, newly developed applications for managing WBEM-capable Net PC platforms should be written to access those platforms through industry-standard WBEM protocols and/or interfaces.

26. WBEM-based service provider enabled on system
Required

The CIMOM protocol and the CIM and Win32 extensions schema will be provided automatically with WMI-instrumented Windows operating systems. For Net PC systems that are pre-installed with these operating systems, these providers must be enabled when the systems are shipped.

DMI-based Management Information Provider

Having a DMI-based management information provider allows a Net PC to be managed by applications that can access the platform through the DMI Management Interface (MI).

To be guaranteed to be able to manage DMI-instrumented Net PCs compliant with these guidelines, management applications must comply with the procedural version of the MI as specified for DMI version 2.0. This includes compliance with *Service Provider API for Management Applications* and *Management Provider API*. The new procedural interface introduced with DMI version 2.0 is an interface that supports remote access using Remote Procedure Call (RPC).

27. DMI service provider present and configured in system

Required

On each DMI-instrumented Net PC, a DMI 2.0 service provider must be present and configured to run whenever the operating system is running. This supports the ability of any compliant DMI management application to access and manage a DMI-instrumented Net PC as soon as the system is up and running.

Simple Network Management Protocol

Managing a Net PC in a corporate enterprise might require that these devices integrate with current enterprise management applications. For this reason, SNMP support can be provided in addition to DMI and WBEM providers.

28. SNMP support in addition to DMI or WBEM providers

Optional

It is recommended that if SNMP support is provided, the same information available through DMI or WMI interfaces should also be available through SNMP.

Industrial Design Requirements

This section summarizes physical design requirements for Net PC systems. Actual form factors are OEM design choices.

29. Minimum set of user indicators

Required

The minimum indicator is a power light. Indicators for hard-disk activity and LAN activity are optional and can be placed for access by service personnel only, with no requirement for end users to be able to view indicators.

30. End user can easily control power through switches and software

Required

The power control buttons must be implemented to meet PC 97 requirements for hardware support for the OnNow design initiative, as defined in item #5 in the "Basic PC 97" chapter of *PC 97 Hardware Design Guide,* plus changes and clarifications published on http://www.microsoft.com/hwdev/pc97.htm.

31. Minimal footprint

Recommended

In general, the form factor chosen for the Net PC should be small and should accommodate the full processor range.

32. Lockable or sealed-case design with no end-user accessible internal expansion capabilities

Required

Net PC systems include no internal expansion slots that are accessible to end users. The Net PC goals do not allow end users to add devices using traditional internal expansion capabilities. The meaning of this requirement is the equivalent of "no user-serviceable parts inside" for consumer electronics or appliances. It does not preclude providing internal expansion slots to allow the OEM to provide additional devices in manufacturing or to allow qualified service personnel to install internal devices.

See also the requirements and recommendations for external expansion buses in the "System Buses" section later in this document.

33. Thermal sensor for monitoring temperature in the chassis

Optional

This solution must be implemented under the ACPI specification, version 1.0 or higher. If implemented in any Net PC system, this must provide support for automatic shutdown in overtemp conditions.

For information about requirements for remote management capabilities, see the "Platform Management Information Requirements" section earlier in this document and the related attachments.

General Device Requirements

This section summarizes requirements for the devices and peripherals provided with Net PC systems.

34. Device driver and installation meet PC 97 requirements for Windows operating systems

Required

Each device supplied with the system must meet PC 97 requirements for that device class, and its drivers must be tested by Windows Hardware Quality Labs (WHQL). The manufacturer does not need to supply a driver if a driver provided with the operating system can be used. If the manufacturer supplies drivers, the requirements for installation include:

- All devices and drivers must pass Microsoft WHQL testing.
- All configuration settings are stored in the registry. The driver must not use INI files for configuration settings.
- The correct minidriver or any other manufacturer-supplied files specified in the device's INF must be installed in the correct locations. For manufacturer-supplied files, the vendor must not be identified as Microsoft and all other copyright and version information must be correct for the manufacturer.
- Driver installation and removal uses the Windows-based methods defined in the appropriate Windows or Windows NT DDK. However, any software applications included with the device can be installed using an alternative Windows-based installation method.
- Driver files provided by the vendor must not use the same file names as used by files included in Microsoft operating systems unless specifically agreed upon by Microsoft.
- It must be possible for the device's driver support to be installed without the user being present, with required parameters supplied by way of a script or other mechanism for predefining settings.

For systems that come pre-installed with Windows NT:

- Only 32-bit protected-mode system-level components are installed. No real-mode or 16-bit protected-mode system-level components are provided.

For systems that come pre-installed with either Windows 98 or Windows NT 5.0, the following requirements apply for drivers:

- For any device for which WDM-based support is provided in the operating system, the driver supplied by the manufacturer must be a WDM minidriver.
- Every WDM driver (or minidriver) must support Plug and Play IRPs.

For systems that come pre-installed with Windows 98, the following requirements apply for drivers:

- Every VxD must support Plug and Play messages.
- Where power management support is provided in the operating system, the VxD must implement device power management as defined in the DDK information provided for Windows 98.

For complete details about installation requirements for drivers, see item #22 in the “Basic PC 97” chapter of *PC 97 Hardware Design Guide*.

35. Each device complies with current Plug and Play specifications
Required
Each device provided in a Net PC system must meet the current Plug and Play specifications related to its class, including requirements defined in the ACPI specification and clarifications published for some Plug and Play specifications. This includes requirements for automatic device configuration, resource allocation, and dynamic disable capabilities.

Standard system devices, such as interrupt controllers, timers, keyboard controllers, RTC, DMA controllers 1 and 2 and page registers, and math co-processors, are exempt from this requirement.

For complete information about this requirement, see item #14 in the “Basic PC 97” chapter of *PC 97 Hardware Design Guide.*

For information about requirements for remote management capabilities, see the “Platform Management Information Requirements” section earlier in this document and the related attachments.

36. Unique Plug and Play device ID for each system device and add-on device
Required
Each device connected to an expansion bus must be able to supply its own unique Plug and Play identifier. Each type of bus contains different information for uniquely identifying devices on expansion cards, with guidelines and exceptions defined in *PC 97 Hardware Design Guide*. For complete requirements, see item #15 in the “Basic PC 97” chapter of *PC 97 Hardware Design Guide.*

Methods for asset-number assignment and serial-number assignment for system components are OEM specific, but must be reported using compliant instrumentation technology. For more information, see the “Platform Management Information Requirements” section earlier in this document and the related attachments.

37. Option ROMs meet Plug and Play requirements, for Intel architecture
Required

This requirement applies only for any devices that might use option ROMs on systems based on Intel architecture, whether the device is present on the system board or provided through an expansion card. Related option ROM requirements are also defined for specific bus classes and specific devices, such as SCSI and graphics adapters, respectively. For complete guidelines, see item #16 in the "Basic PC 97" chapter of *PC 97 Hardware Design Guide*.

38. All devices must support correct 16-bit decoding for I/O port addresses
Required

Each device must support a unique I/O port address in the 16-bit address range. This requirement means that, at a minimum, the upper address lines A10 – A15 can be used as device enable so the device does not respond to addresses outside of the 10-bit address range. For complete guidelines, see item #18 in the "Basic PC 97" chapter of *PC 97 Hardware Design Guide*. CardBus controllers and cards, if present, must meet the requirements defined in the "PC Card" chapter of *PC 97 Hardware Design Guide*.

39. Users are protected from connecting devices incorrectly
Required

This requirement is to help ensure that the end user or service support personnel can correctly make the physical connections required for adding a device to the system. For implementation guidelines, see item #20 in the "Basic PC 97" chapter of *PC 97 Hardware Design Guide*.

40. Minimal interaction required to install and configure devices
Required

After physically installing the device, qualified service personnel must not be required to perform any action other than to provide a pointer to a source that contains drivers and other files. For implementation guidelines, see item #21 in the "Basic PC 97" chapter of *PC 97 Hardware Design Guide*.

41. Multifunction add-on devices meet general device requirements for each device
Required

This requirement is provided as a guideline for OEM-installed devices in a Net PC system. When integrated into a Net PC system, multifunction add-on devices must meet the PC 97 requirements referenced earlier in this section for automated software-only settings for device configuration, device drivers, and Windows-based installation. For implementation guidelines, see item #23 in the "Basic PC 97" chapter of *PC 97 Hardware Design Guide*.

42. Standard system board devices use ISA-compatible addresses, for Intel architecture

Required

This includes devices with I/O port addresses within the reserved range 0h through 0ffh.

System Buses

This section defines requirements for buses provided in a Net PC system.

43. Each bus complies with written specifications and PC 97 requirements

Required

Each bus used in the system must meet all the requirements for that bus as defined in Part 3 of *PC 97 Hardware Design Guide*. This includes the requirement to meet the current Plug and Play specifications related to its class, requirements defined in the ACPI specification, and clarifications published for some Plug and Play specifications. This also includes requirements for automatic device configuration, resource allocation, and dynamic disable capabilities.

44. Universal Serial Bus with one USB port, minimum

Required

The USB implementation in the system must meet the requirements defined in USB specifications, plus the additional requirements for PC 97 as defined in the "USB" chapter of *PC 97 Hardware Design Guide*.

45. PCI bus meets PCI 2.1 specifications and PC 97 requirements

Required

If PCI is used in a Net PC system, the PCI bus must meet the requirements defined in PCI version 2.1 or higher, plus the additional requirements for PC 97 as defined in the "PCI" chapter of *PC 97 Hardware Design Guide*.

Exceptions for particular devices are noted in Parts 3 and 4 of *PC 97 Hardware Design Guide*. For example, add-on PCI IDE devices must comply with PCI 2.1 requirements and also must provide Subsystem IDs and Subsystem Vendor IDs, but PCI-to-PCI bridges and core chip sets do not have to provide Subsystem IDs and Subsystem Vendor IDs.

July 1, 1997, is the compliance date for PCI motherboard devices and for PCI add-on adapters for PCI 2.1 Subsystem IDs. For more information, see the guidelines on http://www.microsoft.com/hwdev/busbios/idpnp.htm.

46. Support for high-speed expansion buses meets PC 97 and Net PC requirements, if present

Required

For Net PC systems, internal expansion capabilities that are accessible by the end user are not allowed. However, if implemented in a Net PC system, all high-speed expansion buses and expansion devices must meet the requirements as specified in Parts 3 and 4 of *PC 97 Hardware Design Guide.* This includes requirements for IEEE 1394 and CardBus, if implemented.

The following are specifically required for Net PC systems:

- Any bus that supports hot plugging or a device designed to use such a bus must support adding or removing devices while the system is fully powered.
- Any devices provided as expansion devices must be capable of being remotely disabled, including the capability to disable drives (CD-ROM, floppy drive, and so on), ensuring that control and TCO policies can be realized. For information about requirements for remote management capabilities, see the “Platform Management Information Requirements” section earlier in this document and the related attachments.

47. Device Bay-capable bay and peripherals

Recommended

If implemented in a Net PC system, Device Bay capabilities must meet the following requirements

- A Device Bay controller, compliant with *Device Bay Interface Specification, Version 1.0* and implemented as an ACPI device object on the system board.
- One USB controller and one IEEE 1394 controller to support all Device Bay-capable bays in the system.
- One USB port and one IEEE 1394 port for each Device Bay-capable bay in the system.

Any Device Bay peripherals provided with a Net PC system must meet the following requirements.

- Peripherals compliant with Device Bay Interface Specification, Version 1.0.
- Peripherals interface with either the USB or IEEE 1394 bus, or both.
- Support relevant USB device class specifications.

The following is specifically required for Net PC systems:

- Any devices provided as expansion devices must be capable of being remotely disabled, including the capability to disable drives (CD-ROM, floppy drive, and so on), ensuring that control and TCO policies can be realized. For information about requirements for remote management capabilities, see the "Platform Management Information Requirements" section earlier in this document and the related attachments.

48. ISA bus expansion slots must not be provided
Required
No ISA bus expansion slots are permitted in a Net PC system. The benefits of an ISA-free Net PC system for administrative support personnel include easier and more stable configuration, lower support costs, and improved performance.

This requirement does not preclude the inclusion of embedded ISA devices in a system board design. However, such devices, along with all other system board devices and system boards in general, must fully comply with all Net PC requirements for ACPI, OnNow, and system board device requirements. The end result must always be that all devices on a system board are fully and completely capable of software detection, configuration, and control.

I/O Devices

This section defines the general requirements for I/O devices.

49. Keyboard connection and keyboard
Required
Recommended: USB

The external connection requirements on any PC can also be met using a PS/2-style port or wireless capabilities in the system. A mobile or all-in-one system that has a built-in keyboard must also provide the capability for an external keyboard connection, which can be implemented using a port replicator or a single PS/2-style port with special cabling for both external keyboard and mouse. For complete requirements for keyboard ports and peripherals, see the "Input Components" chapter in *PC 97 Hardware Design Guide*.

50. Pointing-device connection and pointing device
Required
Recommended: USB or other external bus

The external connection requirements on any PC can also be met using a PS/2-style port or wireless capabilities in the system. A mobile or all-in-one system that has a built-in pointing device must also provide the capability for an external mouse connection, which can be implemented using a port replicator or a single PS/2-style port with special cabling for both external keyboard and mouse. A second serial port is not permitted as the external connection for a pointing device. For complete requirements for mouse ports and peripherals, see the "Input Components" chapter of *PC 97 Hardware Design Guide*.

51. Connection for external parallel devices
Optional
If a parallel port is present, it must be implemented as an Extended Capabilities Port (ECP) mode parallel port. For information about requirements for remote management capabilities, see the "Platform Management Information Requirements" section earlier in this document and the related attachments.

52. Connection for external RS-232C or equivalent devices
Optional
If present, the RS-232C serial connection must be implemented using a 16550A serial port or equivalent. For information about requirements for remote management capabilities, see the "Platform Management Information Requirements" section earlier in this document and the related attachments.

53. Wireless capabilities meet PC 97 requirements, if present
Required
If wireless capabilities are included in the system, PC 97 requirements must be met as defined in the "Serial, Parallel, and Wireless Support" chapter of *PC 97 Hardware Design Guide*. For information about requirements for remote management capabilities, see the "Platform Management Information Requirements" section earlier in this document and the related attachments.

54. Network connectivity meets PC 97 requirements and supports remote new system setup

Required

The Net PC system must include I/O device support and system BIOS support for boot devices to allow installation of the operating system, as described in the "BIOS and Remote New System Setup" section earlier in this document. In addition, the manufacturer must ensure the ability to upgrade the network adapter's option ROM using software, for forward compatibility with remote new system setup capabilities.

The network connectivity device must meet related PC 97 requirements for network communications and for the bus to which it is attached as defined in the "Network Communications" chapter and Part 3 of *PC 97 Hardware Design Guide.* The network connectivity device must be capable of remote configuration and control. For information about requirements for remote management capabilities, see the "Platform Management Information Requirements" section earlier in this document and the related attachments.

The PC 97 guidelines define requirements for NDIS 4.0 support. For Net PC systems that come pre-installed with a Microsoft operating system that supports the extensions in NDIS 5.0, the system must include an NDIS 5.0 network adapter driver. A MAC or NDIS 4.0 implementation is not allowed because NDIS 5.0 support is required to take advantage of new operating system capabilities.

55. Communications device meets PC 97 requirements, if present

Required

Modems or other communications devices such as ISDN cards implemented in the Net PC system must meet PC 97 requirements as defined in the "Modems" and "Network Communications" chapters of *PC 97 Hardware Design Guide*. For information about requirements for remote management capabilities, see the "Platform Management Information Requirements" section earlier in this document and the related attachments.

Graphics Adapter and Multimedia Requirements

This section summarizes the Net PC requirements for the graphics adapter and monitor.

56. Display adapter meets PC 97 and Net PC minimum requirements
Required
Recommended: PCI 2.1 or AGP

A Net PC system must contain a graphics adapter that permits a color depth of 16 bits per pixel (bpp), minimum. The following are the minimum requirements:

- Minimum resolution: 800x600x16 bpp for Windows desktop and 640x480x16 bpp double buffered for Microsoft DirectDraw®–based full-screen applications.

  Display RAM requirements are tied directly to the minimum graphics resolution supported by the adapter. The requirement to support double buffering implies 1.5 MB of display RAM. However, PC 97 requirements do not specify minimum display RAM support; rather, the adapter designer can implement any solutions for supporting the minimum resolution requirements.
- Graphics adapters do not use legacy bus.

  For the graphics adapter, the video bus must not use ISA. A higher-performance solution is required to optimize performance of the packed-pixel frame buffer. Possible implementations that meet this requirement can include PCI 2.1 for all system types or the Intel Accelerated Graphics Port version 1.0 (AGP) interface for systems that have Pentium Pro processors. If the graphics adapter uses the PCI bus, it must comply with PCI 2.1 and additional requirements as defined in the “PCI” and “Graphics Adapters” chapters in *PC 97 Hardware Design Guide*.
- System operates normally with default VGA-mode driver.

  The default VGA driver is required for installing the operating system. The adapter must support 4-bit planar VGA mode as described in the Windows 95 DDK.
- Multimonitor/multiple-display adapter support meets PC 97 requirements, if dual adapter capabilities are possible in the Net PC system as provided by the system manufacturer.

For complete information, see items #1–16 in the “Graphics Adapters” chapter of *PC 97 Hardware Design Guide*.

57. Support for NTSC or PAL television output meets PC 97 requirements, if present

Required

Implementing this support is optional. If implemented, this functionality should meet PC 97 guidelines as defined in item #34 in the “Graphics Adapters” chapter of *PC 97 Hardware Design Guide*.

58. Monitor supports DDC 2.0 Level B, EDID, 800x600 minimum, and PC 97 requirements

Required

A monitor designed for or included with a Net PC system must be compliant with Display Data Channel Standard version 2.0, Level B (DDC2B), which defines the communication channel between the display and host system. In addition, the monitor must transmit an Extended Display Identification Data (EDID) structure containing unique ID Manufacturer Name and ID Product Code identifiers, and all required fields as described in Section 3 of EDID Standard 1.0, revision 1.0.

For implementation guidelines, see item #28 in the “Video Components” chapter of *PC 97 Hardware Design Guide* and the changes and clarifications on http://www.microsoft.com/hwdev/desguid/pc97faq.htm.

59. System supports MPEG-1 playback requirements for PC 97 if system has CD-ROM plus multimedia audio and video capabilities

Required

For Net PC systems that include device support for multimedia, operating system support is provided through Microsoft DirectShow™ (formerly ActiveMovie™). The minimum system requirements to support MPEG-1 playback include:

- Audio and video decode performance: 30 frames per second, minimum, as defined in item #13 in the “Video Components” chapter of *PC 97 Hardware Design Guide.*
- Graphics support for color space conversion and arithmetic stretching, including hardware arithmetic stretching and YUV off-screen surfaces for color space conversion, as defined in item #14 in the “Video Components” chapter of *PC 97 Hardware Design Guide.*

60. PC 97 DVD playback requirements, if system includes DVD-Video

Required

All Net PC systems that include DVD-Video support must provide PC 97 playback support for DVD content, including:

- Video support for decoding MPEG-2 Main Profile and Main Level video streams.
- Audio support for MPEG-2 and Dolby AC-3 decoding and DVD audio mixed with other PC audio streams.
- Synchronized audio and video, meeting general broadcast industry guidelines.
- Independent audio/video streams supported by the decode subsystem.
- WDM-based implementation for MPEG-2 acceleration.

For implementation guidelines, see item #15 in the “Video Components” chapter of *PC 97 Hardware Design Guide.*

61. Audio support meets PC 97 requirements, if present

Required

Recommended: USB or host-based digital audio

If audio capabilities are implemented in a Net PC system, audio must meet PC 97 requirements, whether implemented as baseline audio (items #3–#7) or advanced audio (items #8–#11), as defined in the “Audio Components” chapter of *PC 97 Hardware Design Guide*.

Storage and Related Components

This section presents the requirements and recommendations for storage and related peripheral devices for Net PC systems. See also system BIOS requirements to support high-capacity drives in the “BIOS and Remote New System Setup” section earlier in this guide.

62. Host controller meets PC 97 requirements

Required

Minimum requirements for SCSI, ATA and ATAPI, and IEEE 1394 are defined in related chapters in Part 3 of *PC 97 Hardware Design Guide*.

63. Primary host controller and devices support bus mastering

Required

The primary host controller must support bus mastering, whether using IDE, SCSI, or IEEE 1394. Bus-mastering support must also be enabled for IDE devices, including hard disks, CD-ROM, and tape drives. Bus-master capabilities must meet the related specification for the particular controller. For example, the programming register set for PCI IDE bus master DMA is defined in Small Form Factor (SFF) 8038i.

64. Hard drive meets PC 97 requirements

Required

A hard disk drive is required for the Net PC. The hard drive must meet the PC 97 requirements for hard drives and for the bus it uses, including:

- Drive spin-up time supports OnNow capabilities, as defined in item # 23 in the "Storage and Related Peripherals" chapter of *PC 97 Hardware Design Guide*.
- If IDE is used, each IDE drive must support Master, Slave, and Cable Select settings, as defined in item #24 in the "Storage and Related Peripherals" chapter of *PC 97 Hardware Design Guide*.

65. Hard drive is SMART-compliant

Required

SMART-compliant drive uses SMART IOCTL API, as defined in item #24 in the "ATA and ATAPI" chapter of *PC 97 Hardware Design Guide*.

66. CD-ROM meets PC 97 and Net PC requirements, if present

Required

Recommended: CD-ROM not included in system

If this device is present, the host controller must meet PC 97 requirements, as defined in the related chapter in Part 3 of *PC 97 Hardware Design Guide*. The CD-ROM drive must meet the PC 97 requirements defined in items #25–#30 in the "Storage and Related Components" chapter of *PC 97 Hardware Design Guide*.

For Net PC, the device must be capable of remote lockdown and boot device selection. For information about requirements for remote management capabilities, see the "Platform Management Information Requirements" section earlier in this document and the related attachments.

67. Media status notification support for ATAPI removable media, if present

Required

IDE and ATAPI removable devices must follow the Microsoft specification named Media Status Notification, version 1.03 or higher (included in SFF 8070i).

CD-ROM and DVD-ROM manufacturers must use the Media Status Notification specification contained within the Mt. Fuji specification, which will be provided as SFF 8090 by the SFF Committee.

68. Legacy floppy-disk controller

Optional

Recommended: Floppy disk capabilities not included in system

If a legacy FDC is included on a Net PC system, the drive and controller must meet PC 97 requirements, as defined in items #17–21 in the “Storage and Related Peripherals” chapter of *PC 97 Hardware Design Guide*.

For any implementation of a floppy drive on a Net PC system, the floppy drive must be capable of being remotely disabled (as a boot selection) and must make provisions for locking. For information about requirements for remote management capabilities, see the “Platform Management Information Requirements” section earlier in this document and the related attachments. For information about locking removable media, see the “Hardware Security Features” section later in this document.

Hardware Security Features

This section summarizes the system security requirements and recommendations for Net PC systems. For information about security for limiting user access in preboot modes, see the “BIOS and Remote New System Setup” section earlier in this document.

69. Smart card support meets interoperability specifications, if present

Required

Smart card readers and cards must be compatible with *Interoperability Specification for ICCs and Personal Computer Systems*, published by CP8 Transac, Hewlett-Packard, Microsoft, Schlumberger, and Siemens Nixdorf on http://www.smartcardsys.com.

In addition, smart card readers and device drivers must be Plug and Play compliant and adhere to the Win32 smart card specifications published in the Windows NT DDK. Smart card applications and service provider DLLs must adhere to the Win32 smart card specifications as published in the Win32 SDK.

70. Physical system security

Required

The following security features are required for Net PC hardware to prevent unauthorized access to hardware:

- External drive devices have security capabilities. Each removable media device on a Net PC system must be capable of being secured to prevent unauthorized access to data. This means that the device is rendered useless, either electronically or mechanically.
- PC case and switches have locking capabilities to prevent unauthorized internal access—an OEM-specific method can be implemented, either electronically or mechanically. Usability controls such as volume, brightness, and contrast that are usually configured by the end user are exempt from this requirement.
- Remote software management is supported for physical components. For information about requirements for remote management capabilities, see the “Platform Management Information Requirements” section earlier in this document and the related attachments.

References and Resources

The following presents information resources, services, and tools available to help build Net PC hardware. This section also lists technical references for the specifications cited in these requirements.

Information Resources

Intel developer information
http://developer.intel.com

Information on Net PC, Zero Administration Windows, and hardware development on Microsoft web sites
http://www.microsoft.com/windows/
http://www.microsoft.com/hwdev/

Windows Hardware Quality Labs (WHQL)
whqlinfo@microsoft.com
http://www.microsoft.com/hwtest/

Microsoft Developer Network (MSDN) Professional membership
Phone: (800) 759-5474
Outside North America: (510) 275-0763
Fax: (510) 275-0762
http://www.microsoft.com/msdn/

WBEM information
http://wbem.freerange.com
http://www.microsoft.com/management/wbem/

WMI information
http://www.microsoft.com/management/wbem/

Information on DMTF web site
General information http://www.dmtf.org
Common Information Model (CIM) http://www.dmtf.org/work/cim.html
Specifications http://www.dmtf.org/tech/specs.html

Technical References

Advanced Configuration and Power Interface Specification, Revision 1.0
http://www.teleport.com/~acpi/

ATA 2 [X3T9.2 948D]
ATA Packet Interface for CD-ROM, SFF 8070I and other specifications
Small Computer Interface (SCSI-2) [X3T9.2-375R]
Small Computer Interface (SCSI-3) Parallel Interface (SPI) [X3T9.2/91-10]
Global Engineering Documents
Fax: (303) 397-2740
Telephone: (800) 854-7179; Outside North America: (303) 792-2181

CIM Specifications
http://www.dmtf.org/work/cim.html

Desktop Management Interface Specification, Version 2.00
DMI Compliance Guidelines, Version 1.0
http://www.dmtf.org/tech/specs.html

Device Bay Interface Specification, Version 1.0
http://www.device-bay.org

Device Class Power Management Specifications
http://www.microsoft.com/hwdev/onnow.htm

El Torito—Bootable CD-ROM Format Specification, version 1.0
Compaq, Intel, Phoenix BIOS Boot Specification, version 1.01
http://www.ptltd.com/techs/specs.html

IBM Personal System/2 Common Interfaces, Part No. S84F-9809
IBM Personal System/2 Mouse Technical Reference, Part No. S68X-2229
International Business Machines Corporation
IBM Customer Publications Support: (800) 879-275
Or contact an IBM sales representative

IEEE 1394 Standards
Telephone: (800) 949-4333
Fax: (410) 259-5045
Released Standards: Global Engineering Documents

Intel/Duracell Smart Battery System Specification
http://developer.intel.com/ial/powermgm/specs.htm

International Color Consortium
ICC Profile Format Specification
http://www.color.org

Interoperability Specification for ICCs and Personal Computer Systems
http://www.smartcardsys.com

NLX Motherboard Specification, version 1.0
http://www.teleport.com/~nlx/

Media Status Notification, version 1.03 or higher (included in SFF 8070i)
Mt. Fuji Specification (SFF 8090)
Global Engineering Documents

Multi-session Compact Disc Specification
Enhanced Music CD Specification, version 1.0
Philips Consumer Electronics B.V.
Coordination Office Optical−Magnetic Media Systems
Building SWA-109, PO Box 80002
5600 JB Eindhoven, The Netherlands
Fax: (31) (40) 732113

PC 97 Hardware Design Guide
http://www.microsoft.com/hwdev/pc97.htm

PCI, Version 1.0
PCI Bus Power Management Interface Specification
http://www.pcisig.com

PCMCIA Standards
Personal Computer Memory Card International Association
2635 North First Street, Suite 209
San Jose, CA 95134 USA

Plug and Play specifications
http://www.microsoft.com/hwdev/specs/pnpspecs.htm

Universal Serial Bus, Version 1.0
http://www.usb.org

Universal Serial Bus PC Legacy Compatibility Specification, Version 0.9
USB device class specifications
http://www.teleport.com/~usb/data/usb_le9.pdf

WBEM Specifications
http://wbem.freerange.com

WMI Specifications and Win32 Extensions Schema
http://www.microsoft.com/management/wbem/

Checklist for Network PC Requirements

General System Requirements

1. Minimum CPU: 133-MHz Intel Pentium processor or compatible processor with similar performance, or Windows NT–compatible RISC-based processor
Required

2. Level 2 cache with 256K minimum, for systems with Pentium or compatible processors
Required

3. Minimum RAM: 16 MB
Required

4. Upgrade capabilities for RAM and CPU
Optional

BIOS and Remote New System Setup

5. Limit user access in preboot modes
Required

6. System BIOS support for boot devices, for Intel architecture
Required

7. Support Int 13h Extensions in system BIOS and option ROMs, for Intel architecture
Required

8. BIOS boot support for USB keyboard, if USB is the only keyboard
Required

9. Remote new system setup and service boot supported using DHCP and TFTP as defined in Attachment A
Required

10. Preboot execution environment
Required

11. Remote BIOS update and revision support
Required

Power Management Requirements

12. ACPI support meets PC 97 requirements
Required

13. Hardware support for OnNow initiative
Required

14. BIOS support for OnNow initiative, for Intel architecture
Required

15. Wakeup on LAN supported
Required

Component Instrumentation Requirements

16. Baseline platform management information capabilities
Required

WMI Driver Instrumentation

17. Support WMI/CIM and Win32 extensions schema objects and data
Required

18. Support WMI alert generation for required events
Optional

19. Compliant with WMI alert model for WMI alerts
Required

20. WMI instrumentation interface meets device-specific requirements
Required

DMI Component Instrumentation

21. DMI standard groups instrumented and deployed
Required

22. Components compliant with DMI Component Interface
Required

23. DMI event generation for DMI events in required groups
Optional

24. Compliant with DMI event model for DMI events generated
Required

Management Information Providers

25. At least one management information provider enabled
Required

26. WBEM-based service provider enabled on system
Required

27. DMI service provider present and configured in system
Required

28. SNMP support in addition to DMI or WBEM providers
Optional

Industrial Design Requirements

29. Minimum set of user indicators
Required

30. End user can easily control power through switches and software
Required

31. Minimal footprint
Recommended

32. Lockable or "sealed case" design with no end-user accessible internal expansion capabilities
Required

33. Thermal sensor for monitoring temperature in the chassis
Optional

General Device Requirements

34. Device driver and installation meet PC 97 requirements for Windows operating systems
Required

35. Each device complies with current Plug and Play specifications
Required

36. Unique Plug and Play device ID for each system device and add-on device
Required

37. Option ROMs meet Plug and Play requirements, for Intel architecture
Required

38. All devices must support correct 16-bit decoding for I/O port addresses
Required

39. Users are protected from incorrectly connecting devices
Required

40. Minimal interaction required to install and configure devices
Required

41. Multifunction add-on devices meet general device requirements for each device
Required

42. Standard system board devices use ISA-compatible addresses, for Intel architecture
Required

System Buses

43. Each bus complies with written specifications and PC 97 requirements
Required

44. Universal Serial Bus with one USB port, minimum
Required

45. PCI bus meets PCI 2.1 specifications and PC 97 requirements
Required

46. Support for high-speed expansion buses meets PC 97 and Net PC requirements, if present
Required

47. Device Bay-capable bay and peripherals
Recommended

48. ISA bus expansion slots must not be provided
Required

I/O Devices

49. Keyboard connection and keyboard
Required

50. Pointing-device connection and pointing device
Required

51. Connection for external parallel devices
Optional

52. Connection for external RS-232C devices
Optional

53. Wireless capabilities meet PC 97 requirements, if present
Required

54. Network connectivity meets PC 97 requirements and supports remote new system setup
Required

55. Communications device meets PC 97 requirements, if present
Required

Graphics Adapter and Multimedia Requirements

56. Display adapter meets PC 97 and Net PC minimum requirements
Required

57. Support for NTSC or PAL television output meets PC 97 requirements, if present
Required

58. Monitor supports DDC 2.0 Level B, EDID, 800x600 minimum, and PC 97 requirements
Required

59. System supports MPEG-1 playback requirements for PC 97, if system has CD-ROM plus multimedia audio and video capabilities
Required

60. PC 97 DVD playback requirements, if system includes DVD-Video
Required

61. Audio support meets PC 97 requirements, if present
Required

Storage and Related Components

62. Host controller meets PC 97 requirements
Required

63. Primary host controller and devices support bus mastering
Required

64. Hard drive meets PC 97 requirements
Required

65. Hard drive is SMART-compliant
Required

66. CD-ROM meets PC 97 and Net PC requirements, if present
Required

67. Media status notification support for ATAPI removable media, if present
Required

68. Legacy floppy-disk controller
Optional

Hardware Security Features

69. Smart card support meets Interoperability Specifications, if present
Required

70. Physical system security
Required

Attachment A: DHCP Extensions for New System Setup

This description assumes a knowledge of the standard DHCP/BOOTP protocols.

Protocol Overview

The protocol is a combination of a straightforward extension of DHCP (through the use of several new DHCP Option tags) and the definition of simple packet transactions which use the DHCP packet format and options to pass additional information between the client and server. This added complexity is introduced by the requirement to operate without disturbing existing DHCP services.

In this protocol, DHCP options fields are used to do the following:

- Distinguish DHCP request packets sent by a client as part of this extended protocol from other DHCP request packets that the installation server may receive.
- Distinguish DHCP reply packets sent by a server as part of this extended protocol from other DHCP reply packets that the client may receive.
- Convey client network adapter type.
- Convey client System ID.
- Convey client system architecture type.

Based on the client network adapter type and system architecture type, the server returns to the client the file name (on the server) of an appropriate executable. The client downloads the specified executable into memory and executes it. How this executable accomplishes the setup of the system is not specified by these guidelines.

This section presents an informal, step-by-step description of the remote new system setup protocol. A detailed description of packet formats and client and server actions appears later in this attachment.

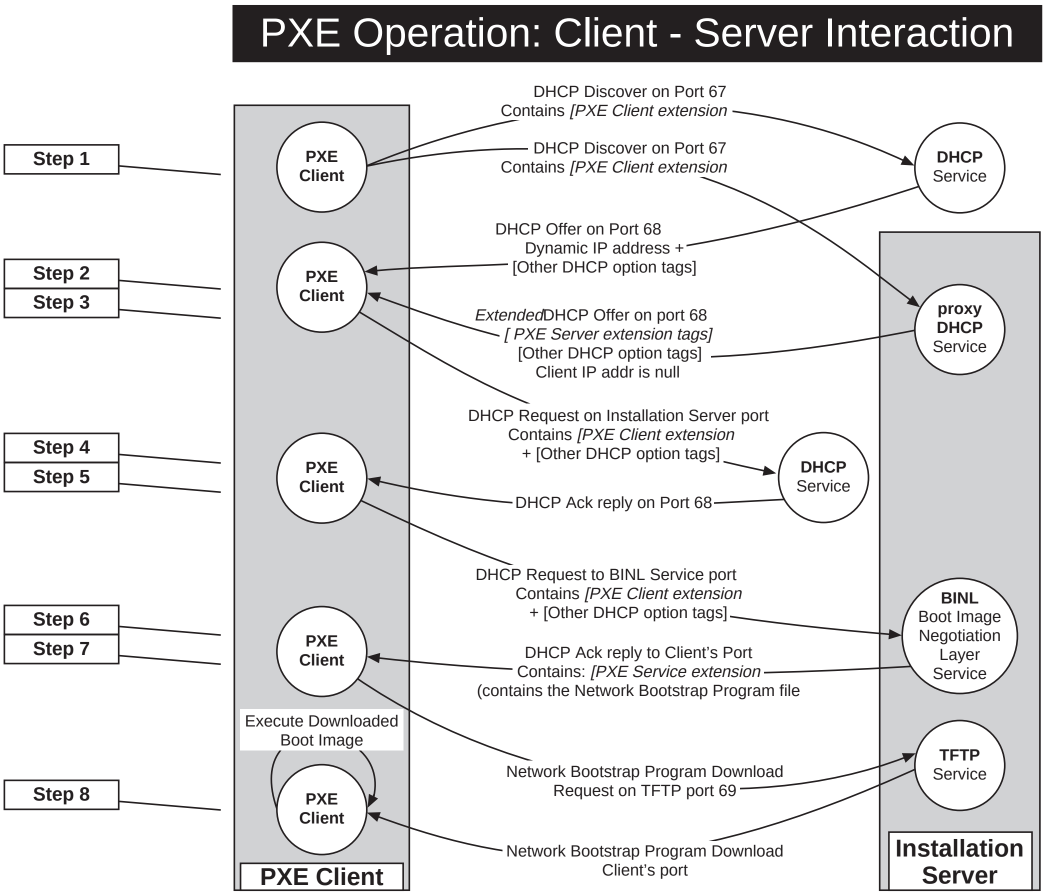


Figure 1 – PXE Operation – Client Server Interaction **(updated in version 1.0b)**

Step 1. The client broadcasts a DHCP discover message to the standard DHCP port (67). An option field in this packet contains the following:

- A tag for client identifier (if the client identifier is known).
- A tag for the client Network Interface Identifier.
- A tag for the client system architecture.

Step 2. The PXE PROXY DHCP Service responds by sending a PXE PROXY DHCPOFFER message to the client on the standard DHCP reply port (68). This packet contains the address of the PXE PROXY DHCP Service. The client IP address field is null.

At this point, other DHCP Services and BOOTP Services also respond with DHCP offers or BOOTP reply messages on port 68. Each message contains standard DHCP parameters: an IP address for the client and any other parameters that the administrator might have configured on the Service. If the Installation Server is also functioning as a standard DHCP Service, then the DHCP Service reply from the Installation Server will also contain standard DHCP parameters (in particular, an IP address for the client)

The timeout for a reply from a DHCP server is standard. The timeout for re-broadcasting to receive a DHCPOFFER with PXE extensions, or a PROXY DHCPOFFER is based on the standard DHCP timeout, but is substantially shorter to allow reasonable operation of the client in standard BOOTP or DHCP environments that do not provide a OFFER with PXE extensions. The PXE timeout for rebroadcast is:

4, 8, 16 seconds, yielding three broadcasts and a timeout after 28 seconds.

The PXE timeout for rebroadcast is 4 seconds after receiving an OFFER without PXE extensions but with a valid “bootfile name” option.

Step 3. From the DHCPOFFER(s) that it receives, the client records the following:

- The Client IP address (and other parameters) offered by a standard DHCP or BOOTP Service.
- The Server IP address of the BINL (Boot Image Negotiation Layer) Service from the “siaddr” field in the PXE proxy DHCP offer.

Step 4. If the client selects an IP address offered by a DHCP Service, then it must complete the standard DHCP protocol by sending a request for the address back to the Service and then waiting for an acknowledgment from the Service. If the client selects an IP address from a BOOTP reply, it can simply go ahead and use the address.

Step 5. The client sends a DHCP Request packet to the BINL Service on port 4011. This packet is exactly the same as the initial DHCP Discover in Step 1, except that it is coded as a DHCP Request and now contains the following:

- Contains the IP address assigned to the client from a DHCP Service.
- Contains all the PXE options fields received from the selected DHCP Offer which contained the PXE options.

Step 6. The BINL Service on the Installation Server sends a DHCP Acknowledge packet back to the client, also on port 4011. This reply packet contains:

- Boot file name and location.
- The Client UUID/GUID option in the PXE proxy DHCP offer.
- MTFTP[1] configuration parameters.

Step 7. The client downloads the executable file using either standard TFTP or MTFTP. The file downloaded and the placement of the downloaded code in memory is dependent on the client's CPU architecture. (For Intel architecture Net PCs, see Attachment B.)

Step 8. Finally, the PXE Client initiates execution of the downloaded code. The way in which this is done is dependent on the client's CPU type. For Intel architecture Net PC systems, the client code executes a far call to the first location in the code.

Relationship to the Standard DHCP Protocol

The initial phase of this protocol piggybacks on a subset of the DHCP protocol messages to enable the client to discover an installation server, that is, one that delivers executables for new system setup. The client *can* use the opportunity to obtain an IP address, which is the expected behavior, but it is not required. Clients that do obtain an IP address using DHCP or BOOTP must implement the protocol as specified in RFC 1541, even though not all possible messages and states of that protocol are described or mentioned in this protocol specification. The points at which this protocol piggybacks or otherwise interacts with the standard DHCP protocol are also noted.

The second phase of this protocol takes place between the client and an installation server, and uses the DHCP message format simply as a convenient format for communication. This second phase of the protocol is otherwise unrelated to the standard DHCP services.

---

[1] Multicast Trivial File Transfer Protocol, as defined by this document through the use of DHCP encapsulated vendor options.

PXE DHCP Options

Tag Name	*Tag Number*	*Length*	*Type*	*Data Field*
Client UUID/GUID	97	17	20	*per Attachment K*
Client Network Interface Identifier	94	3-9	1 = UNDI 2 = PCI 3 = PNP	Type 1 = Major ver(1), Minor Ver(1) Type 2 = Vendor ID(2), Device ID(2), Class Code(3), Rev(1) Type 3 = EISA Device ID(4), Class Code(3)

Tag Name	*Tag Number*	*Length Field*	*Data Field*
Client System Architecture	93	2	0 = Intel Architecture PC 1 = NEC/PC98 2 = etc.
Class Identifier	60	9	“PXEClient”

Encapsulated Vendor Options (DHCP Option #43)

Tag Name	*Tag Number*	*Length Field*	*Data Field*
DHCP_VENDOR	43	varies	Encapsulated options below
(Multiple DHCP_VENDOR options can be used)			

"PXEClient" Encapsulated Options for DHCP Option #43

Tag Name	*Tag Number*	*Length Field*	*Data Field*
PXE_PAD	0	None	None
MTFP IP Addr	1	4	a0, a1, a2, a3
MTFTP Client UDP	2	2	Port Number
MTFTP Server UDP	3	2	Port Number
MTFTP Start Delay	4	1	
MTFTP Timeout Delay	5	1	
Reserved	6-63	1	
Loader Options	64-127	1	(vendor specific)
Vendor Options	128-254	1	(vendor specific)
PXE_END	255	None	

The *Client UUID/GUID* field specifies a globally unique ID (GUID), retrieved from the client system. Client UUID/GUID must be generated per Attachment K. If the Client does not have a GUID, the DHCP service (or Installation Service) may supply one by returning the option with a valid value. The client must store this value if it can, and if so, must use it in all subsequent DHCP transactions.

The *Client Network Interface Identifier* specifies either the physical Network Interface Adapter or indicates the presence of an API (UNDI, described below) that will support a universal boot loader. The UNDI interface should be supported.

The UNDI type field must have a major version of 2 and a minor version of 0 for this version of the protocol. (Future versions may recognize more tags based on this version number.)

If neither PCI nor PNP information is available then the UNDI interface should be supported. Otherwise, the vendor must create an ad hoc PNP or PCI entry and assume the responsibility of distributing the appropriate NIC driver to PXE Servers.

The *Client System Architecture* identifier specifies the system architecture of the client.

The *Class Identifier* (Option 60) of “PXEClient” is required to assure unambiguous identification of clients meeting this specification.

Encapsulated Options (Option 43) are provided to allow configuration for MTFTP boot file transfers. MTFTP should be implemented in the Client. If provided by the DHCP, proxyDHCP, or BINL service, these options should be used.

MTFTP IP Addr is the multicast IP address the client must use to receive the image file.

MTFTP Client UDP is the port the client must listen on to receive the image file.

MTFTP Server UDP is the port the client must use to communicate with the MTFTP service. The client binds to the MTFTP UDP port and waits for the duration of the MTFTP transmission start delay to receive packets.

MTFTP Start Delay is the timeout to begin receiving image file packets before attempting to become the MTFTP acknowledging client (master client) upon initial connection to the MTFTP service.

MTFTP Timeout Delay is the delay, multiplied by the percentage of the file received, the client must wait before attempting to become the MTFTP acknowledging client (master client) upon cessation of packet transmissions during an ongoing MTFTP transfer.

Client Behavior

This section summarizes client behavior for initiation, discovery reply, installation service request, installation service reply, and executable download and execution.

Sending a PXE (Preboot eXecution Environment) Client message requires the use of DHCP Option fields. All PXE Client packets provide the same extended DHCP information in these options. This includes DHCP Request messages used to communicate with the server to which the PXE Client has been redirected. Other fields and options may be different between the packets, based on the standard DHCP protocol.

Initiation

To initiate the interchange between the client and server, the client broadcasts a DHCPDISCOVER packet to the standard DHCP server UDP port (67). The contents of this message must be as described in RFC 1541 for a DHCPDISCOVER message, with the addition of PXE Client option fields. The format of this option is specified in the following table.

DHCP Header		
Field (length)	*Value*	*Comment*
op (1)	1	Code for BOOTP BOOTREQUEST
htype (1)	*	
hlen (1)	*	
hops (1)	*	
xid (4)	*	
secs (2)	*	
flags (2)	*	
ciaddr (4)	0.0.0.0	PXE client always sets this value to 0.0.0.0
yiaddr (4)	blank	Client's IP address. Provided by server
siaddr (4)	*	Next bootstrap server IP address
giaddr (4)	*	
chaddr (16)	xx-xx-xx-xx-xx-xx-xx-xx	Client's MAC address
sname (64)	*	Can be overloaded if using Opt 66
bootfile (128)	*	Can be overloaded if using Opt 67
99.130.83.99		

DHCP Options				
Tag Name	*Tag Number*	*Length*	*Type*	*Data Field*
Client UUID/GUID	97	17	20	*per Attachment K*
Client Network Device Interface Type	94	3-9	1 = UNDI 2 = PCI 3 = PNP	Type 1 = Major ver(1), Minor Ver(1) Type 2 = Vendor ID(2), Device ID(2), Class Code(3), Rev(1) Type 3 = EISA Device ID(4), Class Code(3)
Tag Name	*Tag Number*	*Length Field*		*Data Field*
DHCP Message Type	53	1		1 = DHCPDISCOVER
Class Identifier	60	9		"PXEClient"
Client System Architecture	93	2		0 = Intel Architecture PC, 1 = NEC/PC98

After sending the DHCPDISCOVER message, the client must be prepared to receive replies as described in the following section.

Discovery Reply

In this state, the client is prepared to receive one or more *extended* DHCPOFFER replies from servers on the standard DHCP client UDP port (68). Sending a PXE Server message requires the use of DHCP Options. The format of these options is specified in the following table.

DHCP Header		
Field (length)	*Value*	*Comment*
op (1)	2	Code for BOOTP REPLY
htype (1)	*	
hlen (1)	*	
hops (1)	*	
xid (4)	*	
secs (2)	*	
flags (2)	*	
ciaddr (4)	0.0.0.0	PXE client always sets this value to 0.0.0.0
yiaddr (4)	a0, a1, a2, a3	Client's IP address. Provided by server
siaddr (4)	a0, a1, a2, a3	Next bootstrap server IP address
giaddr (4)	*	
chaddr (16)	*	Client's MAC address
sname (64)	*	Can be overloaded if using Opt 66
bootfile (128)	*	Can be overloaded if using Opt 67
99.130.83.99		

DHCP Options				
Tag Name	*Tag Number*	*Length*	*Type*	*Data Field*
Client UUID/GUID	97	17	20	*per Attachment K*

Tag Name	*Tag Number*	*Length Field*	*Data Field*
DHCP Message Type	53	1	2 = DHCPOFFER
Server Identifier	54	4	a1, a2, a3, a4
Class Identifier	60	9	"PXEClient"

Encapsulated Vendor Options (DHCP Option #43)			
Tag Name	*Tag Number*	*Length Field*	*Data Field*
DHCP_VENDOR	43	varies	Encapsulated options below
(Multiple DHCP_VENDOR options can be used)			

"PXEClient" Encapsulated Options for DHCP Option #43			
Tag Name	*Tag Number*	*Length Field*	*Data Field*
PXE_PAD	0	None	None
MTFP IP Addr	1	4	a0, a1, a2, a3
MTFTP Client UDP	2	2	Port Number
MTFTP Server UDP	3	2	Port Number
MTFTP Start Delay	4	1	
MTFTP Timeout Delay	5	1	
PXE_END	255	None	

If the responding server does not have an installation capability, it will provide a valid address in *siaddr* to redirect the client to an installation server.

In this state, the client must also be prepared to receive one or more *standard* DHCPOFFER messages from servers. Each of these messages will contain configuration information as specified in RFC 1541. Each *extended* DHCPOFFER message can also contain configuration information as specified in RFC 1541. The presence of such information in an *extended* DHCPOFFER message is indicated by a nonzero value in the client IP address field. Which, if any, of these configurations is used by the client is not defined by this specification. If the client decides to accept one of the configurations offered, then it must engage in further communications with the server as specified in RFC 1541.

To move to the installation server request state, the client must have received at least one *extended* DHCPOFFER message. Beyond this, the criteria for the client exiting this state are not defined by this specification.

Installation Service Request

To enter this state, the client must have an IP address. Also, the client must have received one or more *extended* DHCPOFFER messages and therefore know the IP address of one or more installation servers. The client selects one of these installation servers and sends a DHCPREQUEST message to the server on port 4011. Otherwise the format of this message is the same as an *extended* DHCPDISCOVER. The following table lists the required values in the fields of this message; fields marked with an asterisk contain unspecified values.

DHCP Header

Field (length)	*Value*	*Comment*
op (1)	1	Code for BOOTP BOOTREQUEST
htype (1)	*	
hlen (1)	*	
hops (1)	*	
xid (4)	*	
secs (2)	*	
flags (2)	*	
ciaddr (4)	0.0.0.0	PXE client always sets this value to 0.0.0.0
yiaddr (4)	a0, a1, a2, a3	Client's IP address. Provided by DHCP server
siaddr (4)	a0, a1, a2, a3	server IP address
giaddr (4)	0.0.0.0	
chaddr (16)	*xx-xx-xx-xx-xx-xx-xx-xx	Client's MAC address
sname (64)	*	Can be overloaded if using Opt 66
bootfile (128)	*	Can be overloaded if using Opt 67
99.130.83.99		

DHCP Options

Tag Name	*Tag Number*	*Length*	*Type*	*Data Field*
Client UUID/GUID	97	17	20	*per Attachment K*
Client Network Device Interface Type	94	3-9	1 = UNDI 2 = PCI 3 = PNP	Type 1 = Major ver(1), Minor Ver(1) Type 2 = Vendor ID(2), Device ID(2), Class Code(3), Rev(1) Type 3 = EISA Device ID(4), Class Code(3)

Tag Name	*Tag Number*	*Length Field*	*Data Field*
DHCP Message Type	53	1	3 = DHCPREQUEST
Server Identifier	54	4	a1, a2, a3, a4
Class Identifier	60	9	"PXEClient"
Client System Architecture	93	2	0 = Intel Architecture PC 1 = NEC/PC98.

Installation Service Reply

In this state, the client must be prepared to receive an *extended* DHCPACKNOWLEDGE message from the installation server. The following table lists the required values in the fields of this message.

DHCP Header

Field (length)	*Value*	*Comment*
op (1)	2	Code for BOOTP REPLY
htype (1)	*	
hlen (1)	*	
hops (1)	*	
xid (4)	*	
secs (2)	*	
flags (2)	*	
ciaddr (4)	0.0.0.0	PXE client always sets this value to 0.0.0.0
yiaddr (4)	a0, a1, a2, a3	Client's IP address. Provided by server
siaddr (4)	a0, a1, a2, a3	Next bootstrap server IP address
giaddr (4)	*	
chaddr (16)	*	Client's MAC address
sname (64)	*	Can be overloaded if using Opt 66
bootfile (128)	*	Can be overloaded if using Opt 67

99.130.83.99

DHCP Options

Tag Name	*Tag Number*	*Length*	*Type*	*Data Field*
Client UUID/GUID	97	17	20	*per Attachment K*

Tag Name	*Tag Number*	*Length Field*	*Data Field*
DHCP Message Type	53	1	4 = DHCPACKNOWLEDGE
Server Identifier	54	4	a1, a2, a3, a4

Encapsulated Vendor Options (DHCP Option #43)

Tag Name	*Tag Number*	*Length Field*	*Data Field*
DHCP_VENDOR	43	varies	Encapsulated options below

(Multiple DHCP_VENDOR options can be used)

"PXEClient" Encapsulated Options for DHCP Option #43

Tag Name	*Tag Number*	*Length Field*	*Data Field*
PXE_PAD	0	None	None
MTFP IP Addr	1	4	a0, a1, a2, a3
MTFTP Client UDP	2	2	Port Number
MTFTP Server UDP	3	2	Port Number
MTFTP Start Delay	4	1	
MTFTP Timeout Delay	5	1	
PXE_END	255	None	

The options fields in this message must include the following:

- The Installation Server must direct the client to a TFTP server by responding with *siaddr* filled in. (Usually, the TFTP Server resides on the same machine, so *siaddr* would be set to null.)
- Client UUID/GUID if received from the client
- Server Identifier (address of the responding Installation server)
- Bootfile Name if Option 52 (Option Overload) is used.
- *System architecture* indicating architecture the *bootfile* supports. *System architecture* value must be the same as received by the client to insure proper operation of the *bootfile*.

After receiving this message, the client moves to the executable download state.

Executable Download and Execution

In this state, the client is to download all or some portion of the executable file using the standard TFTP. The portion of the file downloaded and the placement of the downloaded code in memory is dependent on the client's CPU architecture.

For Net PC systems based on Intel Architecture, the entire bootstrap image (up to 32K in size) is downloaded into the client PC starting at location 07C00h. The TFTP/MTFTP session that was used to download the bootstrap image is terminated and the logical network connection to the TFTP server is closed.

After the bootstrap image is downloaded, the TFTP connection is closed and control is passed to the bootstrap image. The way in which this is done is dependent on the client's CPU type. For Net PC systems based on Intel Architecture, the boot ROM code is to execute a far call to location 0:7C00h.

MTFTP Operation

Implementation of MTFTP in the client is strongly recommended. If the server sends MTFTP parameters, then the client should proceed as described in this section. In this case the client goes through three phases: an open, a receive and a close, with an error recovery phase that can be entered at any point.

MTFTP open

1. The network client acquires at least the following information from the BINL reply:
   - Client bootfilename
   - MTFTP Server UDP port number
   - MTFTP Client UDP port number
   - MTFTP multicast IP address
   - MTFTP transmission start delay
   - MTFTP transmission time-out delay
2. Client binds to the MTFTP UDP port and waits for the MTFTP transmission start delay to receive packets. No network traffic is generated.
3. If there is a response, MTFTP packets are collected from the network. The client keeps track of received packets in an internal list.

If no packet is received, the client initiates an MTFTP open to the server.

MTFTP receive

1. In order to find out if a client needs to acknowledge or not, the server sends a unicast TFTP packet to that client. The first packet of a MTFTP transmission is always sent both as unicast and multicast UDP/IP. This instructs the network client that it is the acnowledging client.
2. A server always transmits the complete file. Therefore, clients that start listening to a conversation part way through can wait and then get the rest on the next MTFTP transmission to make up for what was missed the first time.
3. The acknowledging client must ACK all packets even if the client has received the entire file.

MTFTP close

1. An MTFTP transmission is finished when the acknowledging client has received all packets and disconnects from the network. Clients who did not receive all packets can initiate a new transmission, if one has not already started.
2. Before a new transmission is started there is a calculated delay. The default delay is modified by an algorithm based on the number of packets received. Clients who received fewer packets will wait for a shorter time than those who received more. This algorithm ensures that:
   - Slow clients define the transmission speed.
   - Clients with a large number of received packets can disconnect from MTFTP after they received all missing packets.
   - Clients who hook into an ongoing MTFTP transmission and therefore only receive the tail of the transmission can disconnect from MTFTP after they received the missing head of the transmission.
   - Clients with a small number of received packets are more likely to become the acknowledging client.

Server Behavior

The server behavior needed for the extended protocol comprises two pieces of functionality: a redirection service, and an installation service.

- The redirection service receives extended DHCPDISCOVER messages (generated by the client Initiation step) on the standard DHCP server port (67) and responds with DHCPOFFER messages containing the location (IP address) of the installation service.
- The installation service receives extended DHCPREQUEST messages (generated by the client Installation Service Request step) on UDP port 4011 and responds with DHCPACK messages containing the location (IP address) of the TFTP service and the file name of a new system setup executable appropriate to the client.

A standard DHCP service may be extended to include the functionality of either the redirection service or the installation service. In this case, this extended DHCP service must implement all behaviors specified for the service included.

Redirection Service Behavior

This section summarizes the behavior of the redirection service to the DHCPDISCOVER message and other DHCP messages.

Response to DHCPDISCOVER

The redirection service will always be prepared to receive on UDP port 67, an extended DHCPDISCOVER message with contents as described earlier in the "Initiation" section. The redirection service will only respond to messages which include DHCP Option 60 with the value of "PXEClient".

If the redirection service responds to a message, it will respond by sending to the initiating client a DHCPOFFER message containing options as described earlier in the "Discovery Reply" section:

The "siaddr" field in the reply, if filled in, will be the location of an installation service. If the "siaddr" field is not filled in then the installation service is at the same address as the redirection service.

The client IP address field of the message will be 0.0.0.0.

If the redirection service is also a standard DHCP configuration service, then the DHCPOFFER message sent to the client will be as specified in RFC 1541.

Installation Service Behavior

This section summarizes the behavior of the installation service to the DHCPREQUEST message and TFTP service messages.

Response to DHCPREQUEST

The installation service will always be prepared to receive a DHCPREQUEST message with contents as described earlier in the "Installation Service Request" section. The installation service will respond by sending to the initiating client a DHCPACKNOWLEDGE message as described earlier in the "Installation Service Reply" section. The file name in this message will be the complete path name of a new system setup executable appropriate to the client that is accessible using TFTP from the installation server's IP address.

TFTP Service

The server running the installation service will provide TFTP service, as described in the previous section.

Attachment B: Preboot Execution Environment

To enable the interoperability of clients and downloaded bootstrap programs, the client preboot code must provide a set of services for use by a downloaded bootstrap. It also must ensure certain aspects of the client state at the point in time when the bootstrap begins executing. The services provided by the client for use by the bootstrap are as follows:

- **Preboot Services API.** Contains several global control and information functions.
- **TFTP API.** Enables opening and closing of TFTP connections, and reading packets from and writing packets to a TFTP connection.
- **UDP API.** Enables opening and closing UDP connections, and reading packets from and writing packets to a UDP connection.
- **Universal NIC Driver Interface (UNDI) API.** Enables basic control of and I/O through the client's network interface device.

The aspects of the client's state to be ensured by the client preboot code at the point in time that execution of the downloaded bootstrap is initiated are as follows:

- The use of certain portions of the client's main memory
- The settings of certain portions of the client's interrupt vector
- The settings of certain of the client's CPU registers

Note: The descriptions in subsequent sections are specific to Intel-architecture PCs. A processor architecture-independent description of these interface and state specifications is probably possible, but has not been attempted.

Client State at Bootstrap Execution Time

This section describes the client state, including information about the bootstrap calling convention, memory usage, and interrupt vector table.

Bootstrap Calling Convention

The entire bootstrap image is downloaded into memory starting at location 07C00h. The preboot code transfers control to the bootstrap by executing a far call to the beginning of the bootstrap code. At this point the following must be true:

- CS:IP is to contain the value 0:7C00h
- ES:BX is to contain the address of the PXENV Entry Point structure described in the “Preboot API Entry Point and Installation Check” section later in this attachment
- EDX is to contain the physical address of the PXENV Entry Point structure
- SS:SP is to contain the address of the beginning of the unused portion of the preboot services stack

Note that the bootstrap code can determine how much free stack space is available by examining the contents of SP and by having knowledge of the memory usage conventions described in the following section.

Caution: A bootstrap should not exceed 32 KB in length. The memory between 07C00h and 10000h is free for use by the bootstrap.

Memory Usage

The following table describes the usage of the first megabyte of the client’s main memory when execution of the downloaded bootstrap is initiated.

Memory Usage During Execution of Downloaded Bootstrap

Address	Status	Preboot services usage	Conventional usage
0 3FF	RESERVED, except for Vector 1Ah at 0:68h.	Vector 1Ah is used to export the preboot services API.	Interrupt Vector Table
400 4FF	RESERVED, except for the 16-bit word at 40:13h	The 16-bit word at 40:13h is the size of free base memory in KB (SFBM/400h).	BIOS Data Area
500 6FF	RESERVED		DOS Data Area
700 7BFF	RESERVED		IO.SYS Load Area
7C00 10000 10000 10000+SFBM-1 10000+SFBM (SS:SP)		Downloaded Bootstrap Free base memory Preboot Services CPU Stack (unused)	
(SS:SP)+1 9FFFF	RESERVED	Preboot Services CPU Stack (used by Preboot Services) Preboot Services Code and Data Extended BIOS Data Area (possibly)	
A0000 BFFFF	RESERVED		Video Memory
C0000 C7FFF	RESERVED		Video BIOS
C8000 DFFFF	RESERVED		Other BIOS / Upper Memory
E0000 EFFFF	RESERVED	Contains a unique system ID structure.	Other BIOS / Upper Memory / System BIOS
F0000 FFFFF	RESERVED	Contains a unique system ID structure.	System BIOS

Free Memory Size (Bios Data Area). When execution of the downloaded bootstrap begins, the 16-bit word at memory address 40:13h must contain the amount of free base memory in KB.

Preboot Services Stack. When execution of the downloaded bootstrap is begun, SS:SP is to contain the address of the top of the unused portion of the preboot services CPU stack. The downloaded image should not modify the used portion of the preboot services CPU stack prior to the time in the boot sequence when it is certain that the preboot services will not be needed again.

Preboot Services Code and Data. This memory area is reserved for the code and data that implement the preboot services. These locations should not be modified by the downloaded image prior to the time in the boot sequence, when it is certain that the preboot services will not be needed again.

Extended BIOS Data. If EBDA has been allocated, the downloaded image should not modify memory in the EBDA.

PXENV Unique System ID (SYSID Bios Area). When execution of the download bootstrap begins, the client's main memory must contain a PXENV unique system ID structure. This structure must meet the following conditions:

- Entry Point Structure **-** This will be found in the 000E0000h to 000FFFFFh physical address area of Memory/RAM. *The Entry Point Structure is PARAGRAPH Aligned.*

Entry Point Structure

Element	**Length**	**Description**
Header/Type	7 Bytes	_SYSID_
Checksum	1 Byte	Checksum of SYSID BIOS Entry Point Structure
Length	2 Bytes	Total length of SYSID BIOS Structure Table (Set to 011h).
SYSID BIOS Structure Table Address	4 Bytes	32 bit physical address of beginning of SYSID BIOS Structure Table. *This value is BYTE Aligned.*
Number of SYSID BIOS Structures	2 Bytes	Total number of structures within the SYSID BIOS Structure Table.
SYSID BIOS Revision	1 Byte	Revision of the SYSID BIOS Extensions (Set to 00h).

UUID Structure Format

Element	Length	Description
Header/Type	6 Bytes	_UUID_
Checksum	1 Byte	Checksum of UUID BIOS Structure
Length	2 Bytes	Total length of UUID BIOS Structure (Set to 0019h).
Variable Data Portion	16 Bytes	Actual UUID data (Initially set all bytes to 0FFh).

1. Header/Type - This is a fixed size for all SYSID BIOS Structure Types. It will always be 6 bytes long. The first and last byte will always be the underscore ascii characters. The middle four bytes are the ASCII characters of UUID.
2. Checksum - This value is a two's complement based checksum which will cause the addition of all bytes defined for this table interface to be equal to 00h. Please note that this is a 8-bit addition calculation (byte wide addition).
3. Length - This value is a Total length of the entire UUID BIOS Structure type. In other words, this value is the addition of all the bytes in this structure from the first byte of the Header/Type field to the last byte in the Variable Data Portion field. The value for this field (for 16 bytes in the Variable Data Portion) is 019h.
4. Variable Data Portion - This value is the 16 byte long (10h) UUID.

Interrupt Vector Table

When execution of the downloaded bootstrap begins, interrupt 1Ah is chained to export the preboot services, TFTP, UDP, and UNDI APIs.

Preboot API Entry Point and Installation Check

Procedures for finding the preboot API entry point structure are architecture dependent. The methods described in this section work for PC/AT x86 clients. In general, the API entry point can be discovered using either of two methods. The first method is to use the installation check interrupt, Int 1Ah. The second is to scan base memory for the preboot API entry point structure. In addition, as described earlier in the "Bootstrap Calling Convention" section, certain registers contain the address of the entry point structure when the downloaded bootstrap is executed.

The preboot API supports only 16-bit real-mode or virtual-86 mode calls. Application programs must make far calls (CALL xxxxh:yyyyh) to the functions in the preboot APIs.

Int 1Ah Function 5650h
(Preboot API Installation Check)

Enter:

AX := 5650h (VP)

Exit:

AX := 564Eh (VN)

ES := 16-bit real-mode segment of the preboot API entry point structure.

BX := 16-bit real-mode offset of the preboot API entry point structure.

EDX := 32-bit physical address of the preboot API entry point structure.

All other register contents are preserved.

CF is cleared.

IF is preserved.

All other flags are undefined.

Preboot API Entry Point Structure

The preboot API entry point structure will be paragraph aligned and placed between the top of free base memory and A0000h (640k). The top of free base memory can be calculated using the size of free base memory word. This word is located in the BIOS data segment at 40:13h.

```
typedef struct s_PXENV_ENTRY {
UINT8 signature[6];      /* "PXENV+" */
UINT16 version;          /* MSB=major, LSB=minor */
UINT8 length;            /* sizeof(struct s_PXENV_ENTRY) */
UINT8 checksum;          /* 8-bit checksum off structure, */
                         /* including this bytes should be 0. */
UINT16 rm_entry_off;     /* 16-bit real-mode offset and segment */
UINT16 rm_entry_seg;     /* of the PXENV API entry point. */
UINT16 pm_entry_off;     /* 16-bit protected-mode offset and */
UINT32 pm_entry_base;    /* segment base address of the */
                         /* PXENV API entry point. */
/* The PROM stack, base code and data segment selectors are only */
     /* required until the base code (TFTP API) layer is removed from */
     /* memory (this can only be done in real mode). */
        UINT16 stack_sel;         /* PROM stack segment.  Will be set */
        UINT16 stack_size;        /* to 0 when removed from memory. */
        UINT16 base_cs_sel;       /* Base code segment.  Will be set */
        UINT16 base_cs_size;      /* to 0 when removed from memory. */
        UINT16 base_ds_sel;       /* Base data segment.  Will be set */
        UINT16 base_ds_size;      /* to 0 when removed from memory. */
     /* The MLID code and data segment selectors are always required */
     /* when running the boot PROM in protected mode. */
        UINT16 mlid_ds_sel;       /* MLID data segment. */
        UINT16 mlid_ds_size;
        UINT16 mlid_cs_sel;       /* MLID code segment. */
        UINT16 mlid_cs_size;
} t_PXENV_ENTRY;
```

Register Usage for Preboot APIs

All API services use the following register settings:

Enter:

BX := PXENV function number

ES := Segment or selector of parameter structure

DI := Offset of parameter structure

Exit:

AX := EXIT_SUCCESS or EXIT_FAILURE

All other register contents are preserved.

CF := Cleared on success, set on error

IF is preserved.

All other flags are undefined.

Preboot Services API

All the fields in the preboot services API parameter structures are to be stored in little endian (Intel) format unless otherwise specified.

UNLOAD PREBOOT STACK

Op-Code: `PXENV_UNLOAD_STACK`

Input: ES:DI points to a `t_PXENV_UNLOAD_STACK` parameter structure.

Output: `PXENV_EXIT_SUCCESS` or `PXENV_EXIT_FAILURE` will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the `PXENV_STATUS_xxx` constants. If successful, the address of the preboot entry point structure will also be filled in.

Description: The preboot services implementation, except for the Universal NIC Driver, will be removed from base memory. UNDI API calls will still be available.

Note: Service cannot be used in protected mode.

Warning! The contents of the preboot entry point structure will be changed by this service. The old preboot entry point structure and contents are invalid and should no longer be used. The CPU stack used by the preboot services will be discarded. The caller must switch to a local CPU stack before making this call. This service should not be used after transferring control to a downloaded OS image.

GET BINL INFO

Op-Code: `PXENV_GET_BINL_INFO`

Input: ES:DI points to a `t_PXENV_GET_BINL_INFO` parameter structure that has been initialized by the caller.

Output: `PXENV_EXIT_SUCCESS` or `PXENV_EXIT_FAILURE` will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the `PXENV_STATUS_xxx` constants. The buffer specified in the parameter structure will be filled with the requested information.

Description: This service will return one of three buffers:

The client's DHCPDISCOVER packet

The DHCP server's DHCPACK packet

The DHCPOFFER packet, which contains Option 60 set to "PXEClient" and a valid bootfile name.

In the downloaded image, the information that is returned by this service is used to configure client ini and cfg files. These files are then used to complete a valid network connection back to the configuration server.

RESTART DHCP

Op-Code: PXENV_RESTART_DHCP

Input: ES:DI points to a t_PXENV_RESTART_DHCP parameter.

Output: If DHCP cannot be restarted, PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants. If DHCP is restarted, control is never returned to the caller.

Description: This service will try to establish a new DHCP connection with the server and try to start a download of a new image. The image to be downloaded will be determined by the server.

Note: It is the responsibility of the caller to make sure the network connection is in a valid state before trying to restart DHCP. Any existing network connection should be closed, and the network adapter must be shutdown using the UNDI API service PXENV_UNDI_SHUTDOWN.

Service cannot be used in protected mode.

RESTART TFTP

Op-Code: PXENV_RESTART_TFTP

Input: ES:DI points to a t_PXENV_RESTART_TFTP parameter structure that has been initialized by the caller. The t_PXENV_RESTART_TFTP parameter structure is identical to the t_PXENV_TFTP_OPEN parameter structure.

Output: If TFTP cannot be restarted, PXENV_EXIT_FAILURE will be returned and CF will be set. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants. If TFTP is restarted, control is never returned to the caller.

Description: This service will try to establish a new TFTP connection with the server and to start a download of a new image. The image to be downloaded will be determined by the previously downloaded image.

Note: It is the responsibility of the caller to make sure the network connection is in a valid state before trying to restart TFTP. The existing network connection with the server needs to be maintained or restored. The existing TFTP connection needs to be closed.

Service cannot be used in protected mode.

MODE SWITCH

Op-Code: `PXENV_MODE_SWITCH`

Input: ES:DI points to a `t_PXENV_MODE_SWITCH` parameter structure that has been initialized by the caller.

Output: The status field in the parameter structure will be set to one of the values represented by the `PXENV_STATUS_xxx` constants.

Description: This service *must* be used when changing the processor between real mode and protected mode. The caller *must* initialize the stack, base code, base data, MLID code, and MLID data selectors and recompute the structure checksum before running this service.

Note: This service can only be called from real mode (before entering, and after leaving, protected mode.) Interrupts need to be disabled when changing the PXENV entry point structure and when calling this service.

Warning! This service can only be used with the default base code interrupt call backs.

TFTP API Service Descriptions

All the fields in the TFTP API parameter structures are to be stored in little endian (Intel) format unless otherwise specified.

TFTP OPEN

Op-Code: `PXENV_TFTP_OPEN`

Input: ES:DI points to a `t_PXENV_TFTP_OPEN` parameter structure that has been initialized by the caller. The IP addresses and port numbers in this structure are to be stored in big endian (Motorola) format.

Output: `PXENV_EXIT_SUCCESS` or `PXENV_EXIT_FAILURE` will be returned in AX, with the CF set accordingly. The `status` field in the parameter structure will be set to one of the values represented by the `PXENV_STATUS_xxx` constants.

Description: Opens a TFTP connection for reading/writing. At any one time there can be only one open connection. The connection must be closed before another can be opened.

TFTP CLOSE

Op-Code: `PXENV_TFTP_CLOSE`

Input: ES:DI points to a `t_PXENV_TFTP_CLOSE` parameter structure that has been initialized by the caller.

Output: `PXENV_EXIT_SUCCESS` or `PXENV_EXIT_FAILURE` will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the `PXENV_STATUS_xxx` constants.

Description: Closes the previously opened TFTP connection.

TFTP READ

Op-Code: PXENV_TFTP_READ

Input: ES:DI points to a t_PXENV_TFTP_READ parameter structure that has been initialized by the caller.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants. When a read is successful, the PacketNumber and PacketLength fields will also be filled in.

Description: Reads one packet from the open TFTP connection.

TFTP/MTFTP READ FILE

Op-Code: PXENV_TFTP_READ_FILE

Input: ES:DI points to a t_PXENV_TFTP_READ_FILE parameter structure that has been initialized by the caller.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants. When a read is successful, the PacketCount and PacketLength fields will also be filled in.

Description: This service will open a TFTP, or MTFTP, connection, download the entire file and close the connection. It is up to the caller to make sure that there is enough free memory to download the file into.

For example, you cannot download a 2 MB file into base memory (below 640K).

Note: UDP open must be called before UDP read or write can be used after transferring a file with this service.

This service cannot be call while in protected mode.

PROTECTED-MODE TFTP/MTFTP READ FILE

Op-Code: PXENV_TFTP_READ_FILE_PMODE

Input: ES:DI points to a t_PXENV_TFTP_READ_FILE_PMODE parameter structure that has been initialized by the caller.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants. When a read is successful, the PacketCount and PacketLength fields will also be filled in.

Description: This service will open a TFTP or MTFTP connection, download the entire file, and close the connection. It is up to the caller to make sure that there is enough free memory to download the file into.

For example, you cannot download a 2-MB file into base memory (below 640K).

Note: UDP open must be called before UDP read or write can be used after transferring a file with this service.

This service cannot be called while in real mode.

TFTP_GET_FILE_SIZE

Op-Code: PXENV_TFTP_GET_FSIZE

Input: ES:DI points to a `t_PXENV_TFTP_Get_FSIZE` parameter structure that has been initialized by the caller.

Output: `PXENV_EXIT_SUCCESS` or `PXENV_EXIT_FAILURE` will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the `PXENV_STATUS_xxx` constants. When the call is successful, the `FileSize` field will be filled in.

Description This service will query the server for the size of the given file using tftp option extension protocol. This service will not and hence must not be used to open a tftp connection for the given file.

Note: This service must not be called when there is an outstanding open tftp connection on the file

UDP API Service Descriptions

UDP OPEN

Op-Code: `PXENV_UDP_OPEN`

Input: ES:DI points to a `t_PXENV_UDP_OPEN` parameter.

Output: `PXENV_EXIT_SUCCESS` or `PXENV_EXIT_FAILURE` will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the `PXENV_STATUS_xxx` constants.

Description: Opens a UDP connection for reading and writing. There can only be one open connection at a time.

UDP CLOSE

Op-Code: `PXENV_UDP_CLOSE`

Input: ES:DI points to a `t_PXENV_UDP_CLOSE` parameter.

Output: `PXENV_EXIT_SUCCESS` or `PXENV_EXIT_FAILURE` will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the `PXENV_STATUS_xxx` constants.

Description: Closes the previously opened UDP connection.

UDP WRITE

Op-Code: `PXENV_UDP_WRITE`

Input: ES:DI points to a `t_PXENV_UDP_WRITE` parameter structure that has been initialized by the caller.

Output: `PXENV_EXIT_SUCCESS` or `PXENV_EXIT_FAILURE` will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the `PXENV_STATUS_xxx` constants.

Description: Writes one packet to the specified IP address on the open UDP connection.

UDP READ

Op-Code: PXENV_UDP_READ

Input: ES:DI points to a t_PXENV_UDP_READ parameter structure that has been initialized by the caller.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: Reads one packet from the opened UDP connection.

UNDI API Service Descriptions

UNDI STARTUP

Op-Code: PXENV_UNDI_STARTUP

Input: ES:DI points to a t_PXENV_UNDI_STARTUP parameter structure that has been initialized by the caller.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: This call provides the Universal NIC Driver with necessary startup parameters, such as the data segment and network adapter identification variables. This call hooks Interrupt 1Ah to export the UNDI API. The rest of the API will not be available until this call has been completed. The data segment must be zero-filled before this API call is made.

Note: The entry point of the UNDI API must be at offset 0 of the UNDI code segment. The preboot code will install the UNDI API by making a far call to the API entry point, with ES:DI and BX setup for UNDI STARTUP.

This service cannot be used in protected mode.

UNDI CLEANUP

Op-Code: PXENV_UNDI_CLEANUP

Input: ES:DI points to a t_PXENV_UNDI_CLEANUP parameter structure.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: This call will uninstall the Interrupt 1Ah hook and will prepare the network adapter driver to be unloaded from memory. This call must be made just before unloading the Universal NIC Driver. The rest of the API will not be available after this call executes.

This service cannot be used in protected mode.

UNDI INITIALIZE

Op-Code: PXENV_UNDI_INITIALIZE

Input: ES:DI points to a t_PXENV_UNDI_INITIALIZE parameter structure that has been initialized by the caller.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: This call resets the adapter and programs it with default parameters. The default parameters used are those supplied to the most recent UNDI_STARTUP call. This routine does not enable the receive and transmit units of the network adapter to readily receive or transmit packets. The application must call PXENV_UNDI_OPEN to logically connect the network adapter to the network.

This call must be made by an application to establish an interface to the network adapter driver. The parameter block to this call contains the pointer to the call-back routines that will be called when a packet is received or when any other interrupt occurs.

Note: When a receive interrupt occurs, the network adapter driver queues the packet and calls the application's callback receive routine with a pointer to the packet received. Then, the callback routine can either copy the packet into its buffer or decide to delay the copy to a later time. The callback receive routine always gets the pointer to the first packet in the receive queue and not to the currently received packet that generated the interrupt.

If the call-back routine decides not to copy the data from the buffer at this time, the packet will remain in the receive queue and, as a result, the later packets might be dropped when the receive queue is full. At a later time, when the application wants to copy the packet, it can call the PXENV_UNDI_FORCE_INTERRUPT routine to simulate the receive interrupt.

When the preboot code makes this call to initialize the network adapter, it passes a NULL pointer for the ProtocolIni field in the parameter structure.

UNDI RESET ADAPTER

Op-Code: PXENV_UNDI_RESET_ADAPTER

Input: ES:DI points to a t_PXENV_UNDI_RESET_ADAPTER parameter structure that has been initialized by the caller.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: This call resets and reinitializes the network adapter with the same set of parameters supplied to Initialize Routine. Unlike Initialize, this call opens the adapter, that is, it connects logically to the network. This routine cannot be used to replace Initialize or Shutdown calls.

UNDI SHUTDOWN

Op-Code: PXENV_UNDI_SHUTDOWN

Input: ES:DI points to a t_PXENV_UNDI_SHUTDOWN parameter.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: This call resets the network adapter and leaves it in a safe state for another driver to program it.

Note: The contents of the PXENV_UNDI_STARTUP parameter structure need to be saved by the Universal NIC Driver in case PXENV_UNDI_INITIALIZE is called again.

UNDI OPEN

Op-Code: PXENV_UNDI_OPEN

Input: ES:DI points to a t_PXENV_UNDI_OPEN parameter structure that has been initialized by the caller.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: This call activates the adapter's network connection and sets the adapter ready to accept packets for transmit and receive.

UNDI CLOSE

Op-Code: PXENV_UNDI_CLOSE

Input: ES:DI points to a t_PXENV_UNDI_CLOSE parameter.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: This call disconnects the network adapter from the network. Packets cannot be transmitted or received until the network adapter is open again.

UNDI TRANSMIT PACKET

Op-Code: PXENV_UNDI_TRANSMIT

Input: ES:DI points to a t_PXENV_UNDI_TRANSMIT parameter structure that has been initialized by the caller.

Output: PXENV_EXIT_SUCCESS or the error code will be returned in AX, with the CF set accordingly. The error code will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: This call transmits a buffer to the network. The media header for the packet can be filled by the calling protocol, but it might not be. The network adapter driver will fill it if required by the values in the parameter block. The transmission is always synchronous and blocks until the network adapter has placed the packet on the network.

UNDI SET MULTICAST ADDRESS

Op-Code: PXENV_UNDI_SET_MCAST_ADDRESS

Input: ES:DI points to a t_PXENV_TFTP_SET_MCAST_ADDRESS parameter structure that has been initialized by the caller.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: This call changes the current list of multicast addresses to the input list and resets the network adapter to accept it. If the number of multicast addresses is zero, multicast is disabled.

UNDI SET STATION ADDRESS

Op-Code: PXENV_UNDI_SET_STATION_ADDRESS

Input: ES:DI points to a t_PXENV_UNDI_SET_STATION_ADDRESS parameter structure that has been initialized by the caller.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: This call sets the MAC address to be the input value and is called before opening the network adapter. Later, the open call uses this variable as a temporary MAC address to program the adapter's individual address registers.

UNDI SET PACKET FILTER

Op-Code: PXENV_UNDI_SET_PACKET_FILTER

Input: ES:DI points to a t_PXENV_UNDI_SET_PACKET_FILTER parameter structure that has been initialized by the caller.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: This call resets the adapter's receive unit to accept a new filter, different from the one provided with the open call.

UNDI GET INFORMATION

Op-Code: PXENV_UNDI_GET_INFORMATION

Input: ES:DI points to a t_PXENV_UNDI_GET_INFORMATION parameter structure that has been initialized by the caller.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: This call copies the network adapter variables, including the MAC address, into the input buffer.

UNDI GET STATISTICS

Op-Code: PXENV_UNDI_GET_STATISTICS

Input: ES:DI points to a t_PXENV_UNDI_GET_STATISTICS parameter structure that has been initialized by the caller.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: This call reads statistical information from the network adapter, and returns.

UNDI CLEAR STATISTICS

Op-Code: PXENV_UNDI_CLEAR_STATISTICS

Input: ES:DI points to a t_PXENV_UNDI_CLEAR_STATISTICS parameter.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: This call clears the statistical information from the network adapter.

UNDI INITIATE DIAGS

Op-Code: PXENV_UNDI_INITIATE_DIAGS

Input: ES:DI points to a t_PXENV_UNDI_INITIATE_DIAGS parameter.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: This call can be used to initiate the run-time diagnostics. It causes the network adapter to run hardware diagnostics and to update its status information.

UNDI FORCE INTERRUPT

Op-Code: PXENV_UNDI_FORCE_INTERRUPT

Input: ES:DI points to a t_PXENV_UNDI_FORCE_INTERRUPT parameter structure that has been initialized by the caller.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: This call forces the network adapter to generate an interrupt. When a receive interrupt occurs, the network adapter driver usually queues the packet and calls the application's callback receive routine with a pointer to the packet received. Then, the callback routine either can copy the packet to its buffer or can decide to delay the copy to a later time. If the packet is not immediately copied, the network adapter driver does not remove it from the input queue. When the application wants to copy the packet, it can call the PXENV_UNDI_FORCE_INTERRUPT routine to simulate the receive interrupt.

UNDI GET MULTICAST ADDRESS

Op-Code: PXENV_UNDI_GET_MCAST_ADDRESS

Input: ES:DI points to a t_PXENV_GET_MCAST_ADDRESS parameter structure that has been initialized by the caller.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the CF set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.

Description: This call converts the given IP multicast address to a hardware multicast address.

UNDI GET NIC TYPE

Op-Code: PXENV_UNDI_GET_NIC_TYPE

Input: ES:DI points to a t_ PXENV_UNDI_GET_NIC_TYPE parameter structure that has been initialized by the caller.

Output: PXENV_EXIT_SUCCESS or PXENV_EXIT_FAILURE will be returned in AX, with the Carry Flag set accordingly. The status field in the parameter structure will be set to one of the values represented by the PXENV_STATUS_xxx constants.
If the PXENV_EXIT_SUCCESS is returned the parameter structure will contain the NIC information.

Description: This call, if successful, provides the NIC specific information necessary to identify the network adapter that is used to boot the system.

Note: The application first gets the DHCP discover packet using GET_BINL_INFO and checks if the UNDI is supported before making this call. If the UNDI is not supported, the NIC specific information can be obtained from the DHCP discover packet itself.

Attachment C: Preboot API Common Type Definitions

Important: The code provided in this attachment is provided for informational purposes only.

```
/*
 *
 * Copyright(c) 1997 by Intel Corporation. All Rights Reserved.
 *
 */

#ifndef _PXENV_CMN_H
#define _PXENV_CMN_H

/* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = */
/* PXENV.H - PXENV/TFTP/UNDI API common, Version 2.x, 97-Jan-17
 *
 * Constant and type definitions used in other PXENV API header files.
 */


/* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = */
/* Parameter/Result structure storage types.
 */
typedef unsigned char UINT8;
typedef unsigned short UINT16;
typedef unsigned long UINT32;
typedef signed char INT8;
typedef signed short INT16;
typedef signed long INT32;


/* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = */
/* Result codes returned in AX by a PXENV API service.
 */
#define PXENV_EXIT_SUCCESS  0x0000
#define PXENV_EXIT_FAILURE  0x0001
#define PXENV_EXIT_CHAIN    0xFFFF  /* used internally */


/* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = */
/* CPU types
 */
#define PXENV_CPU_X86       0
#define PXENV_CPU_ALPHA     1
#define PXENV_CPU_PPC       2
```

```
/* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = */
/* Bus types
 */
#define PXENV_BUS_ISA       0
#define PXENV_BUS_EISA      1
#define PXENV_BUS_MCA       2
#define PXENV_BUS_PCI       3
#define PXENV_BUS_VESA      4
#define PXENV_BUS_PCMCIA    5

/* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = */
/* Status codes returned in the status word of PXENV API parameter
 * structures.
 */

/* Generic API errors that are reported by the loader*/
#define PXENV_STATUS_SUCCESS                                0x00
#define PXENV_STATUS_FAILURE                                0x01
/* General failure. */
#define PXENV_STATUS_BAD_FUNC                               0x02
/* Invalid function number. */
#define PXENV_STATUS_UNSUPPORTED                            0x03
/* Function is not yet supported. */
#define PXENV_STATUS_1A_HOOKED                              0x04
/* Int 1Ah cannot be unhooked. */

/* ARP errors (0x10 to 0x1F) */
#define PXENV_STATUS_ARP_CANCELED_BY_KEYSTROKE              0x10
#define PXENV_STATUS_ARP_TIMEOUT                            0x11

/* BIOS/system errors (0x20 to 0x2F) */
#define PXENV_STATUS_MCOPY_PROBLEM                          0x20

/* TFTP errors (0x30 to 0x3F) */
#define PXENV_STATUS_TFTP_CANNOT_ARP_ADDRESS                0x30
#define PXENV_STATUS_TFTP_OPEN_CANCELED_BY_KEYSTROKE        0x31
#define PXENV_STATUS_TFTP_OPEN_TIMEOUT                      0x32
#define PXENV_STATUS_TFTP_UNKNOWN_OPCODE                    0x33
#define PXENV_STATUS_TFTP_ERROR_OPCODE                      0x34
#define PXENV_STATUS_TFTP_READ_TIMEOUT                      0x35
#define PXENV_STATUS_TFTP_ERROR_OPCODE                      0x36
#define PXENV_STATUS_TFTP_CANNOT_OPEN_CONNECTION            0x38
#define PXENV_STATUS_TFTP_CANNOT_READ_FROM_CONNECTION       0x39
#define PXENV_STATUS_TFTP_TOO_MANY_PACKAGES                 0x3A
#define PXENV_STATUS_TFTP_FILE_NOT_FOUND                    0x3B
#define PXENV_STATUS_TFTP_ACCESS_VIOLATION                  0x3C
#define PXENV_STATUS_TFTP_NO_MCAST_ADDRESS                  0x3D
```

```
/* BOOTP errors 0x40 to 0x4F) */
#define PXENV_STATUS_BOOTP_CANCELED_BY_KEYSTROKE        0x40
#define PXENV_STATUS_BOOTP_TIMEOUT                      0x41
#define PXENV_STATUS_BOOTP_NO_BOOTFILE_NAME             0x43

/* DHCP errors (0x50 to 0x5F) */
#define PXENV_STATUS_DHCP_CANCELED_BY_KEYSTROKE         0x50
#define PXENV_STATUS_DHCP_TIMEOUT                       0x51
#define PXENV_STATUS_DHCP_NO_IP_ADDRESS                 0x52
#define PXENV_STATUS_DHCP_NO_BOOTFILE_NAME              0x53

/* Driver errors (0x60 to 0x6F) */
/* These errors are for UNDI compatible NIC drivers. */
#define PXENV_STATUS_UNDI_MEDIATEST_FAILED              0x61
#define PXENV_STATUS_UNDI_CANNOT_INIT_NIC_FOR_MCAST     0x62

/* Bootstrap (.1) errors (0x70 to 0x7F) */
/* These errors are for the LSA/LCM bootstrap layer. */

/* Environment (.2) errors (0x80 to 0x8F) */
/* These errors are for LSA/LCM environment layers. */

/* MTFTP errors */
#define PXENV_STATUS_MTFTP_OPEN_CANCELED_BY_KEYSTROKE   0x91
#define PXENV_STATUS_MTFTP_OPEN_TIMEOUT                 0x92
#define PXENV_STATUS_MTFTP_UNKNOWN_OPCODE               0x93
#define PXENV_STATUS_MTFTP_READ_CANCELED_BY_KEYSTROKE   0x94
#define PXENV_STATUS_MTFTP_READ_TIMEOUT                 0x95
#define PXENV_STATUS_MTFTP_ERROR_OPCODE                 0x96
#define PXENV_STATUS_MTFTP_CANNOT_OPEN_CONNECTION       0x98
#define PXENV_STATUS_MTFTP_CANNOT_READ_FROM_CONNECTION  0x99
#define PXENV_STATUS_MTFTP_TOO_MANY_PACKAGES            0x9A

/* Misc errors (0xA0 to 0xAF) */
#define PXENV_STATUS_BINL_CANCELED_BY_KEYSTROKE         0xA0
#define PXENV_STATUS_BINL_NO_PXE_SERVER                 0xA1
#define PXENV_STATUS_NOT_AVAILABLE_IN_PMODE             0xA2
#define PXENV_STATUS_NOT_AVAILABLE_IN_RMODE             0xA3
/* Reserved errors (0xB0 to 0xCF) */

/* Vendor errors (0xD0 to 0xFF) */

#endif /* _PXENV_CMN_H */

/* EOF - $Workfile:   pxe_cmn.h  $ */
```

Attachment D: Preboot API Parameter Structure and Type Definitions

Important: The code provided in this attachment is provided for informational purposes only.

```
/*
 *
 * Copyright(c) 1997 by Intel Corporation. All Rights Reserved.
 *
 */

#ifndef _PXENV_API_H
#define _PXENV_API_H

/* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = */
/*  Parameter structure and type definitions for PXENV API version 2.x
 *
 *  PXENV.H needs to be #included before this file.
 *
 *  None of the PXENV API services are available after the stack
 *  has been unloaded.
 */

#include "bootp.h"          /* Defines BOOTPLAYER */


/* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = */
/* Format of PXENV entry point structure.
 */
typedef struct s_PXENV_ENTRY {
    UINT8 signature[6];     /* 'PXENV+' */
    UINT16 version;         /* MSB=major, LSB=minor */
    UINT8 length;           /* sizeof(struct s_PXENV_ENTRY) */
    UINT8 checksum;         /* 8-bit checksum off structure, */
                            /* including this bytes should */
                            /* be 0. */
    UINT16 rm_entry_off;    /* 16-bit real-mode offset and */
    UINT16 rm_entry_seg;    /* segment of the PXENV API entry */
                            /* point. */
    UINT16 pm_entry_off;    /* 16-bit protected-mode offset */
    UINT32 pm_entry_seg;    /* and segment base address of */
                            /* the PXENV API entry point. */
    UINT16 stack_sel;       /* PROM stack segment. Will be set */
    UINT16 stack_size;      /* to 0 when removed from memory. */
```

```
    UINT16 base_cs_sel;      /* Base code segment. Will be set */
    UINT16 base_cs_size;     /* to 0 when removed from memory. */

    UINT16 base_ds_sel;      /* Base data segment. Will be set */
    UINT16 base_ds_size;     /* to 0 when removed from memory. */

    /* The MLID code and data segment selectors are always required */
    /* when running the boot PROM in protected mode. */

    UINT16 mlid_ds_sel;      /* MLID data segment. */
    UINT16 mlid_ds_size;

    UINT16 mlid_cs_sel;      /* MLID code segment. */
    UINT16 mlid_cs_size;

} t_PXENV_ENTRY;

#define PXENV_ENTRY_SIG          "PXENV+"


/* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = */
/* One of the following command op-codes needs to be loaded into the
 * op-code register (BX) before making a call a PXENV API service.
 */
#define PXENV_UNLOAD_STACK      0x70
#define PXENV_GET_BINL_INFO     0x71
#define PXENV_RESTART_DHCP      0x72
#define PXENV_RESTART_TFTP      0x73
#define PXENV_MODE_SWITCH       0x74


/* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = */
/* PXENV API parameter structure typedefs.
 */

/* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = */
typedef struct s_PXENV_UNLOAD_STACK {
    UINT16 status;           /* Out: See PXENV_STATUS_xxx */
                             /*   constants. */
    UINT16 rm_entry_off;     /* Out: 16-bit real-mode segment and */
    UINT16 rm_entry_seg;     /*   offset of PXENV Entry Point */
                             /*   structure. */
    UINT16 pm_entry_off;     /* Out: 16-bit protected-mode offset */
    UINT32 pm_entry_base;    /*   and segment base address of */
                             /*   PXENV Entry Point structure. */
} t_PXENV_UNLOAD_STACK;
```

```
/* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = */
/* Packet types that can be requested in the s_PXENV_GET_BINL_INFO
 * structure. */
#define PXENV_PACKET_TYPE_DHCP_DISCOVER  1
#define PXENV_PACKET_TYPE_DHCP_ACK       2
#define PXENV_PACKET_TYPE_BINL_REPLY     3

/* Three packets are preserved and available through this interface:
 * 1) The DHCP Discover packet sent by the client, 2) the DHCP
 * acknowledgement packet returned by the DHCP server, and 3) the reply
 * packet from the BINL server. If the DHCP server provided the image
 * bootfile name, the DHCP_ACK and BINL_REPLY packets will identical.
 */

/* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = */
typedef struct s_PXENV_GET_BINL_INFO {
    UINT16 status;          /* Out: See PXENV_STATUS_xxx */
                            /*   constants. */
    UINT16 packet_type;     /* In: See PXENV_PACKET_TYPE_xxx */
                            /*   constants */
    UINT16 buffer_size;     /* In: Size of the buffer in */
                            /*   bytes. Specifies the maximum */
                            /*   amount of data that will be */
                            /*   copied by the service. A size */
                            /*   of zero is valid. */
                            /* Out: Amount of BINL data, in */
                            /*   bytes, that was copied into */
                            /*   the buffer. For an input */
                            /*   size of zero, no data will be */
                            /*   copied and buffer_size will */
                            /*   be set to the maximum amount */
                            /*   of data available to be */
                            /*   copied. */
    UINT16 buffer_offset;   /* In: 16-bit offset and segment */
    UINT16 buffer_segment;  /*   selector of a buffer where the */
                            /*   requested packet will be */
                            /*   copied. */
                            /* Out: If buffer_size, buffer_offset */
                            /*   and buffer_segment are all zero; */
                            /*   buffer_offset and buffer_segment */
                            /*   will be changed to point at the */
                            /*   packet buffers in the base code. */
} t_PXENV_GET_BINL_INFO;
```

```
/* - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - */
typedef struct s_PXENV_RESTART_DHCP {
   UINT16 status;            /* Out: See PXENV_STATUS_xxx */
                             /*   constants. */
} t_PXENV_RESTART_DHCP;


/* - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - */
#define s_PXENV_RESTART_TFTP s_PXENV_TFTP_READ_FILE
#define t_PXENV_RESTART_TFTP t_PXENV_TFTP_READ_FILE


typedef struct s_PXENV_MODE_SWITCH {
   UINT16 status;            /* Out: See PXENV_STATUS_xxx constants*/
   UINT16 pxenv_entry_off;   /* In: Offset of PXENV entry point */
                             /*   structure. */
   UINT16 pxenv_entry_seg;   /* In: Real-mode segment or protected- */
                             /*   mode selector of the PXENV */
                             /*   entry point structure. */

    /* Protected-mode status call-back API is documented below. */

   UINT16 pmode_status_off; /* In: Offset of 16-bit protected */
                             /*   mode status call-back. */
   UINT16 pmode_status_sel; /* In: Selector of 16-bit protected */
                             /*   mode status call-back. */
} t_PXENV_MODE_SWITCH;

/*
 * The protected-mode call back will be used by the base code when the
 * client PC is in protected-mode and pmode_status_sel is non-zero.
 *
 * The base code will call the status call-back
 * with the following registers:
 *      AX = 0 (Inside a time-out loop.)
 *      AX = 1 - n (Packet number of received TFTP packet.)
 *      All other registers and flags are undefined.
 *
 * The call-back will return a continue/cancel flag
 * in the following registers:
 *      AX = 0 (continue)
 *      AX = 1 (cancel)
 * All other AX values are undefined, and will be treated as cancel.
 * All other registers and flags must be unchanged.
 */} t_PXENV_MODE_SWITCH;

#endif /* _PXENV_API_H */

/* EOF - $Workfile:  pxe_api.h $ */
```

Attachment E: TFTP API Parameter Structure and Type Definitions

Important: The code provided in this attachment is provided for informational purposes only.

```
/*
 * Copyright(c) 1997 by Intel Corporation. All Rights Reserved.
 *
 */

/* TFTP_API.H
 *  Parameter structure and type definitions for TFTP API version 2.x
 *
 *  PXENV.H needs to be #included before this file.
 *
 *  None of the TFTP API services are available after the stack
 *  has been unloaded.
 */

#ifndef _TFTP_API_H
#define _TFTP_API_H


#include "pxe_cmn.h"


/* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = */
/* #defines and constants
 */

/* One of the following command op-codes needs to be loaded into the
 * op-code register (BX) before making a call a TFTP API service.
 */
#define PXENV_TFTP_OPEN             0x20
#define PXENV_TFTP_CLOSE            0x21
#define PXENV_TFTP_READ             0x22
#define PXENV_TFTP_READ_FILE        0x23
#define PXENV_TFTP_READ_FILE_PMODE  0x24
#define PXENV_TFTP_GET_FSIZE        0x25
```

```
/* Definitions of TFTP API parameter structures.
 */

typedef struct s_PXENV_TFTP_OPEN {
    UINT16 Status;              /* Out: See PXENV_STATUS_xxx */
                                /*      constants. */
    UINT8 ServerIPAddress[4];   /* In:  32-bit server IP */
                                /*      address. Big-endian. */
    UINT8 GatewayIPAddress[4];  /* In:  32-bit gateway IP */
                                /*      address. Big-endian. */
    UINT8 FileName[128];
    UINT16 TFTPPort;            /* In:  Socket endpoint at */
                                /*      which the TFTP server is */
                                /*      listening to requests. */
                                /*      Big-endian. */
} t_PXENV_TFTP_OPEN;


typedef struct s_PXENV_TFTP_GET_FSIZE {
    UINT16 Status;              /* Out: See PXENV_STATUS_xxx */
                                /*      constants. */
    UINT8 ServerIPAddress[4];   /* In:  32-bit server IP */
                                /*      address. Big-endian. */
    UINT8 GatewayIPAddress[4];  /* In:  32-bit gateway IP */
                                /*      address. Big-endian. */
    UINT8 FileName[128];
    UINT32 FileSize;            /* Out: File Size */
} t_PXENV_TFTP_GET_FSIZE;


typedef struct s_PXENV_TFTP_CLOSE {
    UINT16 Status;              /* Out: See PXENV_STATUS_xxx */
                                /*      constants. */
} t_PXENV_TFTP_CLOSE;
```

```
typedef struct s_PXENV_TFTP_READ {
    UINT16 Status;              /* Out: See PXENV_STATUS_xxx */
                                /*      constants. */
    UINT16 PacketNumber;        /* Out: 16-bit packet number. */
    UINT16 BufferSize;          /* In:  Size of the receive */
                                /*      buffer in bytes. */
                                /* Out: Size of the packet */
                                /*      written into the buffer. */
    UINT16 BufferOffset;        /* In:  Segment/Selector and */
    UINT16 BufferSegment;       /*      offset of the receive buffer. */
                                /* Out: Unchanged */
} t_PXENV_TFTP_READ;


typedef struct s_PXENV_TFTP_READ_FILE {
    UINT16 Status;              /* Out: See PXENV_STATUS_xxx */
                                /*      constants. */
    UINT8 FileName[128];        /* In:  file to be read */
    UINT32 BufferSize;          /* In:  Size of the receive */
                                /*      buffer in bytes. */
                                /* Out: Size of the file */
                                /*      written into the buffer. */
    UINT32 BufferOffset;        /* In: 32-bit physical address of the */
                                /*     buffer to load the file into. */
    UINT8 ServerIPAddress[4];   /* In: 32-bit server IP */
                                /*     address. Big-endian. */
    UINT8 GatewayIPAddress[4];  /* In: 32-bit gateway IP */
                                /*     address. Big-endian. */
    UINT8 McastIPAddress[4];    /* In: 32-bit multicast IP address */
                                /*     on which file can be received */
                                /*     can be null for unicast */
    UINT16 TFTPClntPort;        /* In: Socket endpoint on the Client */
                                /*     at which the file can be */
                                /*     received in case of Multicast */
    UINT16 TFTPSrvPort;         /* In: Socket endpoint at which */
                                /*     server listens for requests. */
    UINT16 TFTPOpenTimeOut;     /* In: Timeout value in seconds to be */
                                /*     used for receiving data or ACK */
                                /*     packets. If zero, default */
                                /*     TFTP-timeout is used. */
    UINT16 TFTPReopenDelay;     /* In: wait time in seconds to delay */
                                /*     a reopen request in case of */
                                /*     multicast. */
} t_PXENV_TFTP_READ_FILE;
```

```
typedef struct s_PXENV_TFTP_READ_FILE_PMODE {
    UINT16 Status;              /* Out: See PXENV_STATUS_xxx */
                                /*      constants. */
    UINT8 FileName[128];        /* In: file to be read */
    UINT32 BufferSize;          /* In: Size of the receive */
                                /*     buffer in bytes. */
                                /* Out: Size of the file */
                                /*      written into the buffer. */
    UINT32 BufferOffset;        /* In: 32-bit physical address of the */
                                /*     buffer to load the file into. */
    UINT16 BufferSelector;      /* In: This field must be set to 0 in */
                                /*     real-mode, and to a valid data */
                                /*     selector in protected-mode. */
    UINT8 ServerIPAddress[4];   /* In: 32-bit server IP */
                                /*     address. Big-endian. */
    UINT8 GatewayIPAddress[4];  /* In: 32-bit gateway IP */
                                /*     address. Big-endian. */
    UINT8 McastIPAddress[4];    /* In: 32-bit multicast IP address */
                                /*     on which file can be received */
                                /*     can be null for unicast */
    UINT16 TFTPClntPort;        /* In: Socket endpoint on the Client */
                                /*     at which the file can be */
                                /*     received in case of Multicast */
    UINT16 TFTPSrvPort;         /* In: Socket endpoint at which */
                                /*     server listens for requests. */
    UINT16 TFTPOpenTimeOut;     /* In: Timeout value in seconds to be */
                                /*     used for receiving data or ACK */
                                /*     packets. If zero, default */
                                /*     TFTP-timeout is used. */
    UINT16 TFTPReopenDelay;     /* In: wait time in seconds to delay */
                                /*     a reopen request in case of */
                                /*     multicast. */
} t_PXENV_TFTP_READ_FILE_PMODE;

/* Note:
    If the McastIPAddress specifies a non-zero value, the TFTP_ReadFile
    call tries to listen for multicast packets on the TFTPClntPort
    before opening a TFTP/MTFTP connection to the server.
    If it receives any packets (and not all) or if does not receive any,
    it waits for specified time and tries to reopen a multicast
    connection to the server.
    If the server supports multicast, it notifies the acknowledging
    client with a unicast and starts sending (multicast) the file.
    If the multicast open request times out, the client tries to connect
    to the server at TFTP server port for a unicast transfer.
 */

#endif /* _TFTP_API_H */

/* EOF - $Workfile:  tftp_api.h $ */
```

Attachment F: UDP API Constant and Type Definitions

Important: The code provided in this attachment is provided for informational purposes only.

```
/*
 *
 * Copyright(c) 1997 by Intel Corporation. All Rights Reserved.
 *
 */

#ifndef _UDP_API_H
#define _UDP_API_H


#include "pxe_cmn.h"


/* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = */
/* #defines and constants
 */

#define PXENV_UDP_OPEN  0x30
#define PXENV_UDP_CLOSE 0x31
#define PXENV_UDP_READ  0x32
#define PXENV_UDP_WRITE 0x33


/* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = */
/* Typedefs
 */

typedef struct s_PXENV_UDP_OPEN {
    UINT16 status;          /* Out: See PXENV_STATUS_xxx #defines. */
    UINT8 src_ip[4];        /* Out: 32-bit IP address of this station */

} t_PXENV_UDP_OPEN;
```

```
typedef struct s_PXENV_UDP_CLOSE {
    UINT16 status;          /* Out: See PXENV_STATUS_xxx #defines. */
} t_PXENV_UDP_CLOSE;


typedef struct s_PXENV_UDP_READ {
    UINT16 status;          /* Out: See PXENV_STATUS_xxx #defines. */
    UINT8 src_ip[4];        /* Out: See description below */
    UINT8 dest_ip[4];       /* In/Out: See description below */
    UINT16 s_port;          /* Out: See description below */
    UINT16 d_port;          /* In/Out: See description below */
    UINT16 buffer_size;     /* In: Size of receive buffer. */
                            /* Out: Length of packet written into */
                            /*      receive buffer. */
    UINT16 buffer_off;      /* In: Segment/Selector and offset */
    UINT16 buffer_seg;      /*     of receive buffer. */
} t_PXENV_UDP_READ;

/*
src_ip: (Output)
======
UDP_READ fills this value on return with the 32-bit IP address
of the sender.

dest_ip: (Input/Output)
=======
If this field is non-zero then UDP_READ will filter the incoming
packets and accept those that are sent to this IP address.

If this field is zero then UDP_READ will accept any incoming
packet and return it's destination IP address in this field.

s_port: (Output)
=======
UDP_READ fills this value on return with the UDP port number
of the sender.

d_port: (Input/Output)
=======
If this field is non-zero then UDP_READ will filter the incoming
packets and accept those that are sent to this UDP port.

If this field is zero then UDP_READ will accept any incoming
packet and return it's destination UDP port in this field.

*/
```

```
#define UDP_READ_ANY_IP      0x0000  /* Accept packets sent to any IP.
*/
#define UDP_READ_CHECK_IP    0x0001  /* Only accept packets sent to a */
                                     /*   specific IP address. */


typedef struct s_PXENV_UDP_WRITE {
    UINT16 status;           /* Out: See PXENV_STATUS_xxx #defines. */
    UINT8 ip[4];             /* In: 32-bit destination IP address. */
    UINT8 gw[4];             /* In: 32-bit Gateway IP address. */
    UINT16 src_port;         /* In: Source UDP port, assigned 2069 */
                             /*     if given 0 */
    UINT16 dst port;         /* In: Destination UDP port */
    UINT16 buffer_size;      /* In: Length of packet in buffer. */
    UINT16 buffer_off;       /* In: Segment/Selector and offset */
    UINT16 buffer_seg;       /*     of transmit buffer. */
} t_PXENV_UDP_WRITE;


#endif /* _UDP_API_H */

/* EOF - $Workfile:  udp_api.h $ */
```

Attachment G: UNDI API Constant and Type Definitions

Important: The code provided in this attachment is provided for informational purposes only.

```
/*
 *
 * Copyright(c) 1997 by Intel Corporation. All Rights Reserved.
 *
 */

#ifndef _UNDI_API_H
#define _UNDI_API_H

/* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = */
/* UNDI_API.H
 *  Parameter structure and type definitions for TFTP API version 2.x
 *
 *  PXENV.H needs to be #included before this file.
 *
 *  All of the UNDI API services are still available after the stack
 *  has been unloaded.
 */


/* One of the following command op-codes needs to be loaded into the
 * op-code register (BX) before making a call a TFTP API service.
 */

#include "pxe_cmn.h"

#define PXENV_UNDI_STARTUP              0x0001
#define PXENV_UNDI_CLEANUP              0x0002
#define PXENV_UNDI_INITIALIZE           0x0003
#define PXENV_UNDI_RESET_NIC            0x0004
#define PXENV_UNDI_SHUTDOWN             0x0005
#define PXENV_UNDI_OPEN                 0x0006
#define PXENV_UNDI_CLOSE                0x0007
#define PXENV_UNDI_TRANSMIT             0x0008
#define PXENV_UNDI_SET_MCAST_ADDR       0x0009
#define PXENV_UNDI_SET_STATION_ADDR     0x000A
#define PXENV_UNDI_SET_PACKET_FILTER    0x000B
#define PXENV_UNDI_GET_INFORMATION      0x000C
#define PXENV_UNDI_GET_STATISTICS       0x000D
#define PXENV_UNDI_CLEAR_STATISTICS     0x000E
#define PXENV_UNDI_INITIATE_DIAGS       0x000F
#define PXENV_UNDI_FORCE_INTERRUPT      0x0010
#define PXENV_UNDI_GET_MCAST_ADDR       0x0011

#define ADDR_LEN        16
#define MAXNUM_MCADDR           8
```

```
/* Definitions of TFTP API parameter structures.
 */

typedef struct s_PXENV_UNDI_MCAST_ADDR {
    UINT16 MCastAddrCount;  /* In: Number of multi-cast */
                            /*     addresses. */
    UINT8 MCastAddr[MAXNUM_MCADDR][ADDR_LEN]; /* In: */
                            /*     list of multi-cast addresses. */
                            /*     Each address can take up to */
                            /*     ADDR_LEN bytes and a maximum */
                            /*     of MAXNUM_MCADDR address can */
                            /*     be provided*/
} t_PXENV_UNDI_MCAST_ADDR;


typedef struct s_PXENV_UNDI_STARTUP {
    UINT16 Status;          /* Out: See PXENV_STATUS_xxx constants. */
    UINT8 BusType;          /* In: NIC bus type. */
    UINT8 AddrType;         /*     0 means DataSeg contains segment */
                            /*     addess for DS; 1 means DataSegAddr */
                            /*     contains 32-bit physical addr for */
                            /*     the data segment. */
    UINT16 DataSeg;         /* Segment address for DS */
    UINT32 DataSegAddr;     /* In: 32-bit physical address */
                            /*     of Universal NIC Driver */
                            /*     data segment. */
    UINT16 DataSegSize;     /* In: Size of data segment in bytes. */
    UINT16 CodeSegSize;     /* In: Size of Code segment in bytes. */
    struct {
        UINT16 BusDevFunc;  /* In: Bus, device and function numbers */
                            /*     of this NIC. -1 if not PCI NIC */
        UINT16 PCI_ds_off;  /* Far pointer to PCI data structure */
        UINT16 PCI_ds_seg;
    } pci;
    struct {
        UINT16 CardSelNum;  /* In: Card select number. */
                            /* -1 for non-PnP BBS device */
        UINT16 PnP_eh_off;  /* Far pointer to PnP expansion header */
        UINT16 PnP_eh_seg;
    } pnp;
} t_PXENV_UNDI_STARTUP;


typedef struct s_PXENV_UNDI_CLEANUP {
    UINT16 Status;          /* Out: See PXENV_STATUS_xxx constants. */
} t_PXENV_UNDI_CLEANUP;
```

```
typedef struct s_PXENV_UNDI_INITIALIZE {
    UINT16 Status;            /* Out: See PXENV_STATUS_xxx constants. */
    UINT32 ProtocolIni;       /* In: See description below */
    UINT16 ReceiveOffset;     /* In: See description below */
    UINT16 ReceiveSegment;    /* In: See description below */
    UINT16 GeneralIntOffset;  /* In: See description below */
    UINT16 GeneralIntSegment; /* In: See description below */
} t_PXENV_UNDI_INITIALIZE;


/* ProtocolIni :

    This is an input parameter and is a 32-bit physical address of
    a memory  copy of the  driver module in  the protocol.ini file
    obtained from the  Protocol Manager  driver(refer to  NDIS 2.0
    specifications).   This parameter  is basically  supported for
    the universal NDIS driver to pass the information contained in
    protocol.ini   file  to  the  NIC   driver  for  any  specific
    configuration of   the   NIC.      (Note   that   the   module
    identification in the  protocol.ini  file  was  done  by  NDIS
    itself.)  This value can be NULL for any other application
    interfacing to the Universal NIC Driver.

    ReceiveOffset,  ReceiveSegment:

    This is a pointer to the receive call-back routine and must be
    a non NULL pointer.   This  routine  will  be  called  in  the
    context  of  the  receive  interrupt  after  switching  to  an
    interrupt stack.  The parameters for the routine are passed in
    the registers which  are -  pointer to  the receive  buffer in
    ES:DI and the length of data in CX.  AX contains the length of
    the media  header starting at ES:DI,  BL contains the protocol
    id (0-unknown,  1-IP,  2-ARP,  3-RARP  and  4-others)  and  BH
    contains receive flag (0-directed/promiscuous, 1-broadcast and
    2-multicast).  It is the call-back routine's responsibility to
    initialize it's own  data segment  before starting  to execute
    and to preserve the contents of all the registers except AX.

    The  call-back can either  process the packet  or postpone the
    processing  to a later time.   It must return  a SUCCESS if it
    either copied the  packet into  its own  buffer or  decided to
    reject the  packet after  examining the  packet contents.   In
    this  case the NIC driver removes  the packet from the receive
    queue and recycles the buffer.  If the call-back does not want
    to look at the packet at this time it can return DELAY and the
    NIC  driver keeps the packet in the queue and will always give
    the  first packet's pointer to the call-back in the subsequent
    interrupts.  This  delay  may  however  cause  the  subsequent
    packets  to be dropped if  the receive queue is  full.
```

```
    If the application  decides  to  process  the  packet  it  had
    delayed  it can force the NIC driver to start the call-back by
    calling ForceInterrupt routine.

    GeneralIntOffset, GeneralIntSegment:

    This is also a pointer to a call back routine and will also be
    called  in  the  context  of  an  interrupt.   However,  this
    interrupt is  not  a  receive  interrupt  and  may  be  for  a
    1)transmit  complete, 2)post processing for a previous receive
    interrupt after releasing the interrupt stack or 3)it may be a
    software  interrupt.  The AX  register contains  the function
    code 1,  2 or 3 accordingly.  If  this routine is called for a
    transmit  complete indication, CX register contains the length
    of the packet transmitted and BX register contains the type of
    transmission  0, 1 or 2 according to 0)if the transmit was for
    a directed packet  (i.e.   neither  a  broadcast  and  nor  a
    multicast),  1)if  it  was  a  broadcast  or  2)if  it  was  a
    multicast.


    Note: This call-back  pointer  must  not  be  NULL.   If  the
    application  does not want to process any of these interrupts,
    a pointer to the routine which just returns the status must be
    provided.
*/


typedef struct s_PXENV_UNDI_SHUTDOWN {
    UINT16 Status;          /* Out: See PXENV_STATUS_xxx constants. */
} t_PXENV_UNDI_SHUTDOWN;


typedef struct s_PXENV_UNDI_RESET {
    UINT16 Status;          /* Out: See PXENV_STATUS_xxx constants. */
    t_PXENV_UNDI_MCAST_ADDR R_Mcast_Buf; /* multicast address list */
                                         /* see note below */
} t_PXENV_UNDI_RESET;

 /* Note: The NIC driver does not remember the multicast
    addresses provided in any call.  So the application must
    provide the multicast address list with all the calls that
    reset the receive unit of the adapter.
 */
```

```
typedef struct s_PXENV_UNDI_OPEN {
    UINT16 Status;          /* Out: See PXENV_STATUS_xxx constants. */
    UINT16 OpenFlag;        /* In: See description below */
    UINT16 PktFilter;       /* In: Filter for receiving */
                            /*     packet. It takes the following */
                            /*     values, multiple values can be */
                            /*     ORed together. */
#define FLTR_DIRECTED  0x0001   /* directed/multicast */
#define FLTR_BRDCST    0x0002   /* broadcast packets */
#define FLTR_PRMSCS    0x0004   /* any packet on LAN */
#define FLTR_SRC_RTG   0x0008   /* source routing packet */

    t_PXENV_UNDI_MCAST_ADDR McastBuffer; /* In: */
                                /* See t_PXENV_UNDI_MCAST_ADDR. */
} t_PXENV_UNDI_OPEN;

/*  OpenFlag:
    This is  an input parameter and is  adapter specific.  This is
    supported  for Universal NDIS 2.0 driver to pass down the Open
    flags provided  by   the  protocol   driver  (See   NDIS  2.0
    specifications).  This can be zero.
*/


typedef struct s_PXENV_UNDI_CLOSE {
    UINT16 Status;          /* Out: See PXENV_STATUS_xxx constants. */
} t_PXENV_UNDI_CLOSE;


#define MAX_DATA_BLKS       8

typedef struct s_PXENV_UNDI_TBD{
    UINT16 ImmedLength;     /* In: Data buffer length in */
                            /*     bytes. */
    UINT16 XmitOffset;      /* 16-bit segment & offset of the */
    UINT16 XmitSegment;     /* immediate data buffer. */
    UINT16 DataBlkCount;    /* In: Number of data blocks. */
    struct DataBlk {
        UINT8 TDPtrType;    /* 0 => 32 bit Phys pointer in TDDataPtr */
                            /* not supported in this version of LSA */
                            /* 1 => seg:offser in TDDataPtr which can */
                            /* be a real mode or 16-bit protected */
                            /* mode pointer */
        UINT8 TDRsvdByte;   /* Reserved, must be zero. */
        UINT16 TDDataLen;   /* Data block length in bytes. */
        UINT32 TDDataPtr;   /* Far pointer to data buffer. */
    } DataBlock[MAX_DATA_BLKS];
} t_PXENV_UNDI_TBD;
```

```
typedef struct s_PXENV_UNDI_TRANSMIT {
    UINT16 Status;          /* Out: See PXENV_STATUS_xxx constants. */
    UINT8   Protocol;       /* See description below */
#define P_UNKNOWN   0
#define P_IP        1
#define P_ARP       2
#define P_RARP      3

    UINT8   XmitFlag;           /* See description below*/
#define XMT_DESTADDR    0x0000  /* destination address given */
#define XMT_BROADCAST   0x0001  /* use broadcast address */
    UINT16 DestAddrOffset;      /* 16-bit segment & offset of the */
    UINT16 DestAddrSegment;     /* destination media address */
                                /* See description below */
    UINT16 TBDOffset;           /* 16-bit segment & offset of the */
    UINT16 TBDSegment;          /* transmit buffer descriptor of type
*/
                                /* XmitBufferDesc   */
    UINT32 Reserved[2];         /* for future use   */
} t_PXENV_UNDI_TRANSMIT;

/*
    Protocol:

    This is the protocol  of  the  upper  layer  that  is  calling
    NICTransmit call.   If the  upper layer  has filled  the media
    header this field must be 0.

    XmitFlag:

    If  this flag is  0, the NIC  driver expects a  pointer to the
    destination media  address in the field  DestMediaAddr.  If 1,
    the   NIC  driver   fills  the   broadcast  address   for  the
    destination.

    DestAddrOffset & DestAddrSegment:

    This  is a pointer to the  hardware address of the destination
    media.  It  can be  null if  the destination  is not  known in
    which case the XmitFlag contains 1 for broadcast.  Destination
    media address  must be  obtained by  the upper  level protocol
    (with  Address Resolution Protocol) and NIC driver does not do
    any address resolution.

*/
```

```
typedef struct s_PXENV_UNDI_SET_MCAST_ADDR {
    UINT16 Status;          /* Out: See PXENV_STATUS_xxx constants. */
    t_PXENV_UNDI_MCAST_ADDR McastBuffer;  /* In: */
                            /* See t_PXENV_UNDI_MCAST_ADDR. */
} t_PXENV_UNDI_SET_MCAST_ADDR;


typedef struct s_PXENV_UNDI_SET_STATION_ADDR {
    UINT16 Status;          /* Out: See PXENV_STATUS_xxx constants. */
    UINT8 StationAddress[ADDR_LEN];  /* new address to be set */
} t_PXENV_UNDI_SET_STATION_ADDR;


typedef struct s_PXENV_UNDI_SET_PACKET_FILTER {
    UINT16 Status;          /* Out: See PXENV_STATUS_xxx constants. */
    UINT8 filter;           /* In: Receive filter value. */
                            /* see t_PXENV_UNDI_OPEN for values */
} t_PXENV_UNDI_SET_PACKET_FILTER;


typedef struct s_PXENV_UNDI_GET_INFORMATION {
    UINT16 Status;          /* Out: See PXENV_STATUS_xxx constants. */
    UINT16 BaseIo;          /* Out: Adapter's Base IO */
    UINT16 IntNumber;       /* Out: IRQ number */
    UINT16 MaxTranUnit;     /* Out: MTU */
    UINT16 HwType;          /* Out: type of protocol at hardware level
*/

#define ETHER_TYPE 1
#define EXP_ETHER_TYPE 2
#define IEEE_TYPE  6
#define ARCNET_TYPE  7
```

```
    /*  other numbers can  be obtained from  rfc1010 for "Assigned
    Numbers".  This number may not be validated by the application
    and hence adding new numbers to the list should be fine at any
    time.  */

    UINT16 HwAddrLen;       /* Out: actual length of hardware address */
    UINT8 CurrentNodeAddress[ADDR_LEN]; /* Out: Current hardware */
                                        /*      address*/
    UINT8 PermNodeAddress[ADDR_LEN];    /* Out: Permanent hardware */
                                        /*      address*/
    UINT16 ROMAddress;      /* Out: ROM address */
    UINT16 RxBufCt;         /* Out: receive Queue length */
    UINT16 TxBufCt;         /* Out: Transmit Queue length */
} t_PXENV_UNDI_GET_INFORMATION;


typedef struct s_PXENV_UNDI_GET_STATISTICS {
    UINT16 Status;          /* Out: See PXENV_STATUS_xxx constants. */
    UINT32 XmtGoodFrames;   /* Out: No. of successful transmissions*/
    UINT32 RcvGoodFrames;   /* Out: No. of good frames received */
    UINT32 RcvCRCErrors;    /* Out: No. of frames with CRC error */
    UINT32 RcvResourceErrors;    /* Out: no. of frames discarded - */
                                 /* Out: receive Queue full */
} t_PXENV_UNDI_GET_STATISTICS;


typedef struct s_PXENV_UNDI_CLEAR_STATISTICS {
    UINT16 Status;          /* Out: See PXENV_STATUS_xxx constants. */
} t_PXENV_UNDI_CLEAR_STATISTICS;


typedef struct s_PXENV_UNDI_INITIATE_DIAGS {
    UINT16 Status;          /* Out: See PXENV_STATUS_xxx constants. */
} t_PXENV_UNDI_INITIATE_DIAGS;


typedef struct s_PXENV_UNDI_FORCE_INTERRUPT {
    UINT16 Status;          /* Out: See PXENV_STATUS_xxx constants. */
} t_PXENV_UNDI_FORCE_INTERRUPT;
```

```
typedef struct s_PXENV_UNDI_GET_MCAST_ADDR {
    UINT16 Status;          /* Out: See PXENV_STATUS_xxx constants. */
    UINT32 InetAddr;        /* In: IP Multicast Address */
    UINT8 MediaAddr[ADDR_LEN];  /* Out: corresponding hardware */
                                /*      multicast address */
} t_PXENV_UNDI_GET_MCAST_ADDR;

#define PXENV_UNDI_GET_NIC_TYPE 0x12

typedef s_PXENV_UNDI_GET_NIC_TYPE{
UINT16 Status;              /* OUT: See PXENV_STATUS_xxx constants */
UINT8 NicType;              /* OUT: 2=PCI, 3=PnP */
union{
    struct{
        UINT16 Vendor_ID;   /* OUT: */
        UINT16 Dev_ID;      /* OUT: */
        UINT8 Base_Class;   /* OUT: */
        UINT8 Sub_Class;    /* OUT: */
        UINT8 Prog_Intf;    /* OUT: program interface */
        UINT8 Rev;          /* OUT: Revision number */
        UINT16 BusDevFunc;  /* OUT: Bus, Device */
                            /*      & Function numbers */
    }pci;
        struct{
            UINT32 EISA_Dev_ID; /* Out: */
            UINT8 Base_Class;   /* OUT: */
            UINT8 Sub_Class;    /* OUT: */
            UINT8 Prog_Intf;    /* OUT: program interface */
            UINT16 CardSelNum;  /* OUT: Card Selector Number */
            }pnp;
    }pci_pnp_info;
}t_PXENV_UNDI_GET_NIC_TYPE;

#endif /* _UNDI_API_H */

/* EOF - $Workfile:  undi_api.h $ */
```

Attachment H: WMI/CIM and Win32 Extensions Instrumentation Details

The WMI/CIM required schema is a set of base classes that provide the minimal set of information supported by Net PC platforms deployed with WMI instrumented operating system. The minimal set of classes and associations are listed here and must be deployed on all Net PCs designed for shipment with WMI-instrumented and WBEM-instrumented operating systems.

Over time, the set of base classes can be extended to accommodate additional requirements. CIM incorporates an extension process whereby additional classes, properties, and associations can be introduced. These will initially be introduced as non-standard extensions, typically subclasses of a standard class or new classes associated with a standard class. Over time, the model is intended to evolve as new extensions become widely used and accepted (clearly, the better the design, the better the chance of an extension being accepted).

It is the intention of the standard set defined here to establish a baseline for Net PC management, not a comprehensive solution to all possible management scenarios. Individual OEMs can extend the schema as required to accommodate special capabilities. If these extensions are made under the class structure defined here, management applications can be expected to benefit from them without any changes being required.

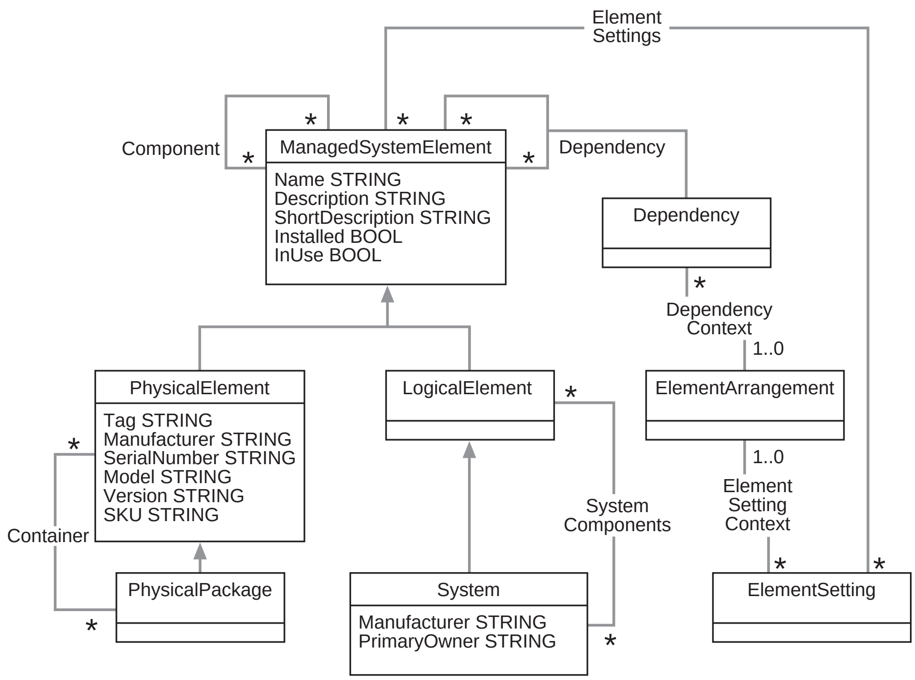
The class structure defined here is a subset of the overall CIM schema. Any extensions are required to be consistent with this broader schema. Extensions outside the logical framework of the CIM schema will require changes to management applications to take full advantage of them. As these extensions are standardized, management applications can be expected to accommodate the new capabilities with any required changes to associated algorithms and interfaces. Generic browsers, of course, can always display a new class or property without requiring any specialized extensions.

Management applications can be expected to take advantage of the CIM schema to provide, for example, different views of a system (a physical component view, a services and drivers view, a running processes view, and so on.)

An example of the extension of a standard class is the addition of a new type of service as an extension of the standard service class. Any management application that uses the standard service class will pick up instances of the new service class (as a result of inheritance), even though it will not be aware of the extension. If someone adds a new class that is not a subclass of the standard service class, yet the instances of the new class are services, then management applications will not be aware of the new class without some exception handling. The management application will have to be specially coded to go and look for the class and to present it along with the other information about services.

A similar but more complex argument applies to the use of associations.

The individual elements that make up a system can be enumerated using a number of different strategies. There are several key classes and associations involved; most are present in the diagram that follows. The strategies available for enumerating the components of a system provide a view of the schema from the perspective of a system considered as an aggregation hierarchy (as opposed to the schema as a classification hierarchy).



A function that lists the components of a system will start with a system object. The system components association relates the system to its components. The listing function must select the components to be listed based on the type of picture of the system to be presented. There is any number of alternative views depending on the circumstances at hand:

- If a list of the physical components is required, the function will list all the components that are physical elements.
- If the top-most physical elements are required, the function will list the physical elements that are not contained in anything.
- If a configuration view is required, the objects representing the physical configuration of the system will be accessed. Its dependency and context associations will be traversed to obtain the physical configuration.
- If the logical components of the system are required, the components of type logical element will be selected.
- Selecting logical elements that have nothing dependent on them would return top-level logical objects.
- Low-level elements—typically, a device-level view—could be constructed by selecting logical elements that either have no realization or have a physical element realization.
- Dependency or configuration trees could be constructed by pursuing suitable associations.

Name	SuperClass	Description
ManagedSystemElement		The base class for all system component objects. Any managed object that is a component of a system is a descendent of this class. These objects include: software components, such as files; devices, such as disk drives and controllers; and physical components, such as chips and cards.
PhysicalElement	ManagedSystemElement	Any component of a system that has a distinct physical identity and that can be defined in terms of labels that can be physically attached to the object is a member of this class. All processes, files, records, and devices are considered not to be physical elements. For example, it is not possible to attach a label to a modem. It is only possible to attach a label to the card that implements the modem. The same card could also implement a LAN adapter. These are tangible managed system elements—usually actual hardware items—that have a physical manifestation of some sort. A managed system element is not necessarily a discrete component. For example, it is possible for a single card—which is a type of physical element—to host more than one logical device. The card would be represented by a single physical element associated with multiple logical devices.
LogicalElement	ManagedSystemElement	A base class for all the components of the system that represent abstract system components, such as profiles, processes, or system capabilities in the form of logical devices.
System	LogicalElement	A grouping of other logical elements. Because systems are logical elements, a system can be composed of other systems.
Protocol	LogicalElement	Represents a protocol, which is a set of rules and algorithms that govern the interaction between two or among more than two interfaces.
SoftwareComponent	LogicalElement	Represents any software component, which can be either an individual file, such as an executable, or a collection of files, such as packages or operating systems. Software components can have additional associated information, such as the installation date.
Process	LogicalElement	A sequence of states defined by the interaction of one or more processors or interpreters, some executable code and a set of inputs.
Thread	LogicalElement	Represents a thread, which is a unit of execution; that is, an address space. Threads are owned by processes.

Name	SuperClass	Description
SystemService	LogicalElement	Represents a system service, which is a definition of a process owned by the system; rather than some specific user that provides an interface to some aspect of the functionality supported by the system.
Job	LogicalElement	Represents a unit of work, such as a print job.
JobDestination	LogicalElement	A process or service able to process one or more jobs.
FileSystem	LogicalElement	This object represents a set of conventions used for arranging data on a storage medium.
DiskPartition	LogicalElement	A structure used to manage the physical surface of a physical disk. There may be a level of indirection between the physical disk and the actual hardware as, for example, in the case of RAID devices.
Device	LogicalElement	A unit of functionality associated with providing the basic capabilities of a system such as input, output, or storage management. Devices may be directly expressed by a physical component as, for example, in the case of a keyboard. However, almost any device can be virtualized either by simulation (for example, simulating a modem using main CPU cycles), by allocation of a single device to multiple physical components, or by allocation of more than one device to a single physical unit. For example, a modem and LAN adapter may share the same PCMCIA card.
StorageDevice	Device	A source or destination for a file. Processors and end users typically see data in the system in terms of files, which in turn are allocated to data sources. The data source is a named unit of storage that may correspond to a variety of implementations, including memory, CDROM, and network.
Modem	Device	A device that translates binary data into wave modulations — typically, sound for transmission over telephone lines.
Processor	Device	A device capable of interpreting a sequence of machine instructions. Typically, the processor has a close correspondence to a physical chip, but it may be provided by an interpreter that is itself a process running on a processor of some kind.
Keyboard	Device	A device for entering data through keystrokes.
LogicalConnector	Device	A device capable of connecting two or more other devices.

Name	SuperClass	Description
InterfaceDevice	Device	Represents an interface device; any interface device is a descendant of this class. These are devices that act as an interface between a device and the rest of the system — for example, disk controllers, serial ports, parallel ports, and so on.
Display	Device	The device used to visually display output from the system.
MemoryModule	Device	A device capable of storing information for fast retrieval.
PointingDevice	Device	A device used to point to regions on the display.
Printer	Device	A device capable of reproducing a visual image on a medium of some kind, usually paper. Printers are a common example of a device that is also a system. The system aspect of the printer must be represented in this model as a discrete object that is a descendent of the system class.
ActualStorageDevice	Device	A device that is primarily intended to describe the organization of a physical unit used to store data.
Bus	Device	A device that provides high-bandwidth communication between different components of the system.
SCSIIInterface	InterfaceDevice	Represents a SCSII Interface device and its properties.
NetworkDrive	StorageDevice	Represents a logical drive that has been mapped to a network resource.
LogicalDiskDrive	StorageDevice	A data source that resolves to a local ActualStorageDevice.
Driver	SoftwareComponent	Represents an executable or set of executables that provide an interface either to another driver or to a logical device.
OperatingSystem	SoftwareComponent	Describes general information about operating systems installed on this system.
NetworkProtocol	Protocol	Provides information about a protocol that has been installed on the system.
ComputerSystem	System	A system that is capable of running programs, processing inputs, and displaying or otherwise returning outputs.
PhysicalPackage	PhysicalElement	Defines the characteristics of system components that physically contain other system components, such as the system enclosure, which would be a type of cabinet.

Name	SuperClass	Description
PhysicalLink	PhysicalElement	Contains any physical object used to link other objects together, which can include wires, wireless connections (radio frequencies and infrared), and so on.
PhysicalConnector	PhysicalElement	A physical element that is used to connect other Physical Elements, such as slots and plugs. This object has properties, such as the type (male or female) and the number of pins.
Slot	PhysicalConnector	Defines the attributes for the different expansion slots supported by this system.
PortConnector	PhysicalConnector	Defines the network connection points provided by the system.
Card	PhysicalPackage	A type of physical container that can be plugged into another card or board.
ElementSetting		ElementSetting are operational parameters that vary from time to time.
PartitionConfiguration	ElementSetting	An arrangement of partitions used to provide a basis for one or more logical disks.
Dependency		An association class that is the base class for all associations that define any dependency between managed system elements.
Component		Descendents of this association class define part of the relationships between managed system elements. For example, the system components association defines the parts of a system.
Location		The base class for all location objects.
ElementSettings		Relates an ElementSetting object to the system element it provides settings for.

Attachment I: DMI Instrumentation Details

The following standard groups from the *System Standards Group Definition, Approved Version 1.0*, must be instrumented and deployed on DMI-instrumented systems compliant with these guidelines:

- DMTF|ComponentID|001
- DMTF|Disk Mapping Table|001
- DMTF|Disks|002
- DMTF|General Information|001
- DMTF|Keyboard|003
- DMTF|Mouse|003
- DMTF|Operating System|001
- DMTF|Partition|001
- DMTF|Physical Container Global Table|001
- DMTF|Processor|003
- DMTF|System BIOS|001
- DMTF|System Cache|002
- DMTF|System Slots|003
- DMTF|Video BIOS|001
- DMTF|Video|002

The following table contains standard groups related to system resource management from the *System Standards Group Definition, Approved Version 1.0*. All these groups are valid standard groups, but the groups designated as “Replacement” groups are designed to replace the two groups marked “Original.” The DMTF recommends that instrumentation migrate to the Replacement groups.

To be compliant with these guidelines, a DMI-instrumented system provides either all of the Original groups or all of the corresponding Replacement groups. It is highly recommended that the Replacement groups be selected for newly implemented instrumentation.

DMI Standard Group	Implementation Guidelines
DMTF|System Resource 2|001	Replacement, recommended for new instrumentation
DMTF|System Resource Device Info|001	Replacement, recommended for new instrumentation
DMTF|System Resource DMA Info|001	Replacement, recommended for new instrumentation
DMTF|System Resource I/O Info|001	Replacement, recommended for new instrumentation
DMTF|System Resource IRQ Info|001	Replacement, recommended for new instrumentation
DMTF|System Resource Memory Info|001	Replacement, recommended for new instrumentation
DMTF|System Resources|001	Original, recommended for legacy instrumentation only
DMTF|System Resources Description|001	Original, recommended for legacy instrumentation only

The following table contains standard groups related to physical memory management from the *System Standards Group Definition, Approved Version 1.0*. All of these groups are valid standard groups, but the groups designated as “Replacement” groups are designed to replace the group marked “Original.” The DMTF recommends that instrumentation migrate to the Replacement groups. To be compliant with these guidelines, a DMI-instrumented system should be instrumented with either the Original group or the corresponding Replacement groups. It is highly recommended that the Replacement groups be selected for newly implemented instrumentation.

DMI Standard Group	Implementation Guidelines
DMTF|Memory Device|001	Replacement, recommended for new instrumentation
DMTF|Memory Array Mapped Addresses|001	Replacement, recommended for new instrumentation
DMTF|Memory Device Mapped Addresses|001	Replacement, recommended for new instrumentation
DMTF|Physical Memory Array|001	Replacement, recommended for new instrumentation
DMTF|Physical Memory|002	Original, recommended for legacy instrumentation only

The following standard groups from the *LAN Adapter Standard Groups Definition, Release Version 1.0*, must be instrumented and deployed on DMI-instrumented systems compliant with these guidelines:

- DMTF|Network Adapter 802 Port|001
- DMTF|Network Adapter Driver|001

Attachment J: Possible DMI/CIM Mappings

The following table is provided as a planning tool and will change as the DMTF completes its work on mapping DMI 2.0 groups into CIM classes. It maps the DMTF groups supported by Net PC platforms to the CIM classes supported by Net PC platforms. Note that there is generally not a one-to-one correspondence, as typically the CIM classes are more normalized than the groups, implying that properties will be spread across more than one class.

DMTF Group	CIM Equivalent
DMTF|ComponentID|001	System and associated objects
DMTF|Disk Mapping Table|001	Association between disks and partitions
DMTF|Disks|002	Elements of logical disk and physical disk
DMTF|General Information|001	System
DMTF|Keyboard|003	Keyboard and keyboard settings
DMTF|Mouse|003	Pointing device and settings
DMTF|Operating System|001	Operating system
DMTF|Partition|001	Disk partition
DMTF|Physical Container Global Table|001	System and associated physical elements
DMTF|Processor|003	Processor
DMTF|System BIOS|001	Win32BIOS and SystemROM
DMTF|System Cache|002	System cache
DMTF|System Slots|003	Logical connector and slot
DMTF|Video BIOS|001	Win32BOS
DMTF|Video|002	Display and display controller

Attachment K: UUIDs and GUIDs

Network Working Group	Paul J. Leach, Microsoft
INTERNET-DRAFT	Rich Salz, Open Group
<draft-leach-uuids-guids-00.txt>	
Category: Informational	
Expires August 24, 1997	February 24, 1997

UUIDs and GUIDs

STATUS OF THIS MEMO

ABSTRACT

This specification defines the format of UUIDs (Universally Unique IDentifier), also known as GUIDs (Globally Unique IDentifier). A UUID is 128 bits long, and if generated according to the one of the mechanisms in this document, is either guaranteed to be different from all other UUIDs/GUIDs generated until 3400 A.D. or extremely likely to be different (depending on the mechanism chosen). UUIDs were originally used in the Network Computing System (NCS) [1] and later in the Open Software Foundation’s (OSF) Distributed Computing Environment [2].

This specification is derived from the latter specification with the kind permission of the OSF.

Contents

Introduction

This specification defines the format of UUIDs (Universally Unique IDentifiers), also known as GUIDs (Globally Unique IDentifiers). A UUID is 128 bits long, and if generated according to the one of the mechanisms in this document, is either guaranteed to be different from all other UUIDs/GUIDs generated until 3400 A.D. or extremely likely to be different (depending on the mechanism chosen).

Motivation

One of the main reasons for using UUIDs is that no centralized authority is required to administer them (beyond the one that allocates IEEE 802.1 node identifiers). As a result, generation on demand can be completely automated, and they can be used for a wide variety of purposes. The UUID generation algorithm described here supports very high allocation rates: 10 million per second per machine if you need it, so that they could even be used as transaction IDs.

UUIDs are fixed-size (128-bits) which is reasonably small relative to other alternatives. This fixed, relatively small size lends itself well to sorting, ordering, and hashing of all sorts, storing in databases, simple allocation, and ease of programming in general.

Specification

A UUID is an identifier that is unique across both space and time, with respect to the space of all UUIDs. To be precise, the UUID consists of a finite bit space. Thus the time value used for constructing a UUID is limited and will roll over in the future (approximately at A.D. 3400, based on the specified algorithm). A UUID can be used for multiple purposes, from tagging objects with an extremely short lifetime, to reliably identifying very persistent objects across a network.

The generation of UUIDs does not require that a registration authority be contacted for each identifier. Instead, it requires a unique value over space for each UUID generator. This spatially unique value is specified as an IEEE 802 address, which is usually already available to network-connected systems. This 48-bit address can be assigned based on an address block obtained through the IEEE registration authority. This section of the UUID specification assumes the availability of an IEEE 802 address to a system desiring to generate a UUID, but if one is not available section 4 specifies a way to generate a probabilistically unique one that can not conflict with any properly assigned IEEE 802 address.

Format

The following table gives the format of a UUID. The UUID consists of a record of 16 octets. The fields are in order of significance for comparison purposes, with “time_low” the most significant, and “node” the least significant.

Field	Data Type	Octet #	Note
time_low	unsigned 32 bit integer	0-3	The low field of the timestamp.
time_mid	unsigned 16 bit integer	4-5	The middle field of the timestamp.
time_hi_and_version	unsigned 16 bit integer	6-7	The high field of the timestamp multiplexed with the version number.
clock_seq_hi_and_reserved	unsigned 8 bit integer	8	The high field of the clock sequence multiplexed with the variant.
clock_seq_low	unsigned 8 bit integer	9	The low field of the clock sequence.
node	unsigned 48 bit integer	10-15	The spatially unique node identifier.

To minimize confusion about bit assignments within octets, the UUID record definition is defined only in terms of fields that are integral numbers of octets. The version number is in the most significant 4 bits of the time stamp (*time_hi*), and the variant field is in the most significant 3 bits of the clock sequence (*clock_seq_high*).

The timestamp is a 60 bit value. For UUID version 1, this is represented by Coordinated Universal Time (UTC) as a count of 100-nanosecond intervals since 00:00:00.00, 15 October 1582 (the date of Gregorian reform to the Christian calendar).

The following table lists currently defined versions of the UUID.

Msb0	Msb1	Msb2	Msb3	Version	Description
0	0	0	1	1	The version specified in this document.
0	0	1	0	2	Reserved for DCE Security version, with embedded POSIX UIDs.

The variant field determines the layout of the UUID. The structure of UUIDs is fixed across different versions within a variant, but not across variants; hence, other UUID variants may not interoperate with the UUID variant specified in this document. Interoperability of UUIDs is defined as the applicability of operations such as string conversion, comparison, and lexical ordering across different systems. The *variant* field consists of a variable number of the msbs of the *clock_seq_hi_and_reserved* field.

The following table lists the contents of the variant field.

Msb0	Msb1	Msb2	Description
0	-	-	Reserved, NCS backward compatibility.
1	0	-	The variant specified in this document.
1	1	0	Reserved, Microsoft Corporation GUID.
1	1	1	Reserved for future definition.

The clock sequence is required to detect potential losses of monotonicity of the clock. Thus, this value marks discontinuities and prevents duplicates. An algorithm for generating this value is outlined in the “Clock Sequence” section below.

The clock sequence is encoded in the 6 least significant bits of the *clock_seq_hi_and_reserved* field and in the *clock_seq_low* field.

The *node* field consists of the IEEE address, usually the host address. For systems with multiple IEEE 802 nodes, any available node address can be used. The lowest addressed octet (octet number 10) contains the global/local bit and the unicast/multicast bit, and is the first octet of the address transmitted on an 802.3 LAN.

Depending on the network data representation, the multi-octet unsigned integer fields are subject to byte swapping when communicated between different endian machines.

The nil UUID is special form of UUID that is specified to have all 128 bits set to 0 (zero).

Algorithms for Creating a UUID

Various aspects of the algorithm for creating a UUID are discussed in the following sections. UUID generation requires a guarantee of uniqueness within the node ID for a given variant and version. Interoperability is provided by complying with the specified data structure. To prevent possible UUID collisions, which could be caused by different implementations on the same node, compliance with the algorithm specified here is required.

Clock Sequence

The clock sequence value must be changed whenever:

- The UUID generator detects that the local value of UTC has gone backward.
- The UUID generator has lost its state of the last value of UTC used, indicating that time *may* have gone backward; this is typically the case on reboot.

While a node is operational, the UUID service always saves the last UTC used to create a UUID. Each time a new UUID is created, the current *UTC* is compared to the saved value and if either the current value is less (the non-monotonic clock case) or the saved value was lost, then the *clock sequence* is incremented modulo 16,384, thus avoiding production of duplicate UUIDs.

The *clock sequence* must be initialized to a random number to minimize the correlation across systems. This provides maximum protection against *node* identifiers that may move or switch from system to system rapidly. The initial value MUST NOT be correlated to the node identifier.

The rule of initializing the *clock sequence* to a random value is waived if, and only if all of the following are true:

- The *clock sequence* value is stored in non-volatile storage.
- The system is manufactured such that the IEEE address ROM is designed to be inseparable from the system by either the user or field service, so that it cannot be moved to another system.
- The manufacturing process guarantees that only new IEEE address ROMs are used.
- Any field service, remanufacturing or rebuilding process that could change the value of the clock sequence must reinitialise it to a random value.

In other words, the system constraints prevent duplicates caused by possible migration of the IEEE address, while the operational system itself can protect against non-monotonic clocks, except in the case of field service intervention. At manufacturing time, such a system may initialise the clock sequence to any convenient value.

System Reboot

There are two possibilities when rebooting a system:

- The UUID generator state—the last UTC, adjustment, and clock sequence—of the UUID service has been restored from non-volatile store
- The state of the last UTC or adjustment has been lost.

If the state variables have been restored, the UUID generator just continues as normal. Alternatively, if the state variables cannot be restored, they are reinitialised, and the clock sequence is changed.

If the clock sequence is stored in non-volatile store, it is incremented; otherwise, it is reinitialised to a new random value.

Clock Adjustment

UUIDs may be created at a rate greater than the system clock resolution. Therefore, the system must also maintain an adjustment value to be added to the lower-order bits of the time. Logically, each time the system clock ticks, the adjustment value is cleared. Every time a UUID is generated, the current adjustment value is read and incremented atomically, then added to the UTC time field of the UUID.

Clock Overrun

The 100 nanosecond granularity of time should prove sufficient even for bursts of UUID creation in high-performance multiprocessors. If a system overruns the clock adjustment by requesting too many UUIDs within a single system clock tick, the UUID service may raise an exception, handled in a system or process-dependent manner either by:

- Terminating the requester
- Reissuing the request until it succeeds
- Stalling the UUID generator until the system clock catches up.

If the processors overrun the UUID generation frequently, additional node identifiers and clocks may need to be added.

UUID Generation

UUIDs are generated according to the following algorithm:

- Determine the values for the UTC-based timestamp and clock sequence to be used in the UUID, as described above.
- For the purposes of this algorithm, consider the timestamp to be a 60-bit unsigned integer and the clock sequence to be a 14-bit unsigned integer. Sequentially number the bits in a field, starting from 0 (zero) for the least significant bit.
- Set the *time_low* field equal to the least significant 32-bits (bits numbered 0 to 31 inclusive) of the time stamp in the same order of significance.
- Set the *time_mid* field equal to the bits numbered 32 to 47 inclusive of the time stamp in the same order of significance.
- Set the 12 least significant bits (bits numbered 0 to 11 inclusive) of the *time_hi_and_version* field equal to the bits numbered 48 to 59 inclusive of the time stamp in the same order of significance.
- Set the 4 most significant bits (bits numbered 12 to 15 inclusive) of the *time_hi_and_version* field to the 4-bit version number corresponding to the UUID version being created, as shown in the table above.
- Set the *clock_seq_low* field to the 8 least significant bits (bits numbered 0 to 7 inclusive) of the *clock sequence* in the same order of significance.
- Set the 6 least significant bits (bits numbered 0 to 5 inclusive) of the *clock_seq_hi_and_reserved* field to the 6 most significant bits (bits numbered 8 to 13 inclusive) of the *clock sequence* in the same order of significance.
- Set the 2 most significant bits (bits numbered 6 and 7) of the *clock_seq_hi_and_reserved* to 0 and 1, respectively.
- Set the *node* field to the 48-bit IEEE address in the same order of significance as the address.

String Representation of UUIDs

For use in human readable text, a UUID string representation is specified as a sequence of fields, some of which are separated by single dashes.

Each field is treated as an integer and has its value printed as a zero-filled hexadecimal digit string with the most significant digit first. The hexadecimal values a to f inclusive are output as lower case characters, and are case insensitive on input. The sequence is the same as the UUID constructed type.

The formal definition of the UUID string representation is provided by the following extended BNF:

```
UUID                    = <time_low> "-" <time_mid> "-"
                          <time_high_and_version> "-"
                          <clock_seq_and_reserved>
                          <clock_seq_low> "-" <node>
time_low                = 4*<hexOctet>
time_mid                = 2*<hexOctet>
time_high_and_version   = 2*<hexOctet>
clock_seq_and_reserved  = <hexOctet>
clock_seq_low           = <hexOctet>
node                    = 6*<hexOctet
hexOctet                = <hexDigit> <hexDigit>
hexDigit =
        "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"
        | "a" | "b" | "c" | "d" | "e" | "f"
        | "A" | "B" | "C" | "D" | "E" | "F"
```

The following is an example of the string representation of a UUID:

```
f81d4fae-7dec-11d0-a765-00a0c91e6bf6
```

Comparing UUIDs

Consider each field of the UUID to be an unsigned integer as shown in the table in section 3.1. Then, to compare a pair of UUIDs, arithmetically compare the corresponding fields from each UUID in order of significance and according to their data type. Two UUIDs are equal if and only if all the corresponding fields are equal. The first of two UUIDs follows the second if the most significant field in which the UUIDs differ is greater for the first UUID. The first of a pair of UUIDs precedes the second if the most significant field in which the UUIDs differ is greater for the second UUID.

Byte order of UUIDs

UUIDs may be transmitted in many different forms, some of which may be dependent on the presentation or application protocol where the UUID may be used. In such cases, the order, sizes and byte orders of the UUIDs fields on the wire will depend on the relevant presentation or application protocol. However, it is strongly RECOMMENDED that the order of the fields conform with ordering set out in section 3.1 above. Furthermore, the payload size of each field in the application or presentation protocol MUST be large enough that no information lost in the process of encoding them for transmission.

In the absence of explicit application or presentation protocol specification to the contrary, a UUID is encoded as a 128-bit object, as follows: the fields are encoded as 16 octets, with the sizes and order of the fields defined in section 3.1, and with each field encoded with the Most Significant Byte first (also known as network byte order).

```
0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          time_high                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       time_mid                |         time_hi_and_version   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|clk_seq_hi_res |  clk_seq_low  |         node (0-1)            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         node (2-5)                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Node IDs when no IEEE 802 network card is available

If a system wants to generate UUIDs but has no IEE 802 compliant network card or other source of IEEE 802 addresses, then this section describes how to generate one.

The ideal solution is to obtain a 47 bit cryptographic quality random number, and use it as the low 47 bits of the node ID, with the most significant bit of the first octet of the node ID set to 1. This bit is the unicast/multicast bit, which will never be set in IEEE 802 addresses obtained from network cards; hence, there can never be a conflict between UUIDs generated by machines with and without network cards.

If a system does not have a primitive to generate cryptographic quality random numbers, then in most systems there are usually a fairly large number of sources of randomness available from which one can be generated. Such sources are system specific, but often include:

- The percent of memory in use
- The size of main memory in bytes
- The amount of free main memory in bytes
- The size of the paging or swap file in bytes
- Free bytes of paging or swap file
- The total size of user virtual address space in bytes
- The total available user address space bytes
- The size of boot disk drive in bytes
- The free disk space on boot drive in bytes
- The current time
- The amount of time since the system booted
- The individual sizes of files in various system directories
- The creation, last read, and modification times of files in various system directories
- The utilization factors of various system resources (heap, etc.)
- Current mouse cursor position
- Current caret position
- Current number of running processes, threads
- Handles or IDs of the desktop window and the active window
- The value of stack pointer of the caller
- The process and thread ID of caller
- Various processor architecture specific performance counters (instructions executed, cache misses, TLB misses)

(Note that it precisely the above kinds of sources of randomness that are used to seed cryptographic quality random number generators on systems without special hardware for their construction.)

In addition, items such as the computer's name and the name of the operating system, while not strictly speaking random, will help differentiate the results from those obtained by other systems.

The exact algorithm to generate a node ID using these data is system specific, because both the data available and the functions to obtain them are often very system specific. However, assuming that one can concatenate all the values from the randomness sources into a buffer, and that a cryptographic hash function such as MD5 [3] is available, the following code will compute a node ID:

```
#include <md5.h>
#define HASHLEN 16

void GenNodeID(
    unsigned char * pDataBuf,   // concatenated "randomness values"
    long cData,                 // size of randomness values
    unsigned char NodeID[6]     // node ID
)
{
  int i, j, k;
  unsigned char Hash[HASHLEN];
  MD_CTX context;

  MDInit (&context);
  MDUpdate (&context, pDataBuf, cData);
  MDFinal (Hash, &context);

  for (j = 0; j<6; j++) NodeId[j]=0;
  for (i = 0,j = 0; i < HASHLEN; i++) {
        NodeID[j++] ^= Hash[i];
        if (j == 6) j = 0;
    };
    NodeID[0] |= 0x80;      // set the multicast bit
};
```

Other hash functions, such as SHA-1 [4], can also be used (in which case HASHLEN will be 20). The only requirement is that the result be suitably random—in the sense that the outputs from a set uniformly distributed inputs are themselves uniformly distributed, and that a single bit change in the input can be expected to cause half of the output bits to change.

Obtaining IEEE 802 addresses

The following URL

http://stdsbbs.ieee.org/products/oui/forms/index.html

contains information on how to obtain an IEEE 802 address block. Cost is $1000 US.

Security Considerations

It should not be assumed that UUIDs are hard to guess; they should not be used as capabilities.

Acknowledgements

This document draws heavily on the OSF DCE specification for UUIDs. Ted Ts'o provided helpful comments, especially on the byte ordering section which we mostly plagiarized from a proposed wording he supplied (all errors in that section are our responsibility, however).

References

[1] Lisa Zahn, et. al., Network Computing Architecture, Prentice Hall, Englewood Cliffs, NJ, 1990

[2] DCE: Remote Procedure Call, Open Group CAE Specification C309 ISBN 1-85912-041-5 28cm. 674p. pbk. 1,655g. 8/94

[3] R. Rivest, RFC 1321, “The MD5 Message-Digest Algorithm”, 04/16/1992.

[4] SHA Spec - TBD

Authors' addresses

Paul J. Leach
Microsoft
1 Microsoft Way
Redmond, WA, 98052, U.S.A.
Email: paulle@microsoft.co

Rich Salz
The Open Group
11 Cambridge Center
Cambridge, MA 02142, U.S.A.
Email r.salz@opengroup.org

UUID Reference Implementation

```
/*
** Copyright (c) 1990- 1993, 1996 Open Software Foundation, Inc.
** Copyright (c) 1989 by Hewlett-Packard Company, Palo Alto, Ca. &
** Digital Equipment Corporation, Maynard, Mass.
** To anyone who acknowledges that this file is provided "AS IS"
** without any express or implied warranty: permission to use, copy,
** modify, and distribute this file for any purpose is hereby
** granted without fee, provided that the above copyright notices and
** this notice appears in all source code copies, and that none of
** the names of Open Software Foundation, Inc., Hewlett-Packard
** Company, or Digital Equipment Corporation be used in advertising
** or publicity pertaining to distribution of the software without
** specific, written prior permission.  Neither Open Software
** Foundation, Inc., Hewlett-Packard Company, nor Digital Equipment
** Corporation makes any representations about the suitability of
** this software for any purpose.
*/
#include <sys/types.h>
#include <sys/time.h>

typedef unsigned long   unsigned32;
typedef unsigned short  unsigned16;
typedef unsigned char   unsigned8;
typedef unsigned char   byte;

#define CLOCK_SEQ_LAST              0x3FFF
#define RAND_MASK                   CLOCK_SEQ_LAST

typedef struct _uuid_t {
    unsigned32          time_low;
    unsigned16          time_mid;
    unsigned16          time_hi_and_version;
    unsigned8           clock_seq_hi_and_reserved;
    unsigned8           clock_seq_low;
    byte                node[6];
} uuid_t;

typedef struct _unsigned64_t {
    unsigned32          lo;
    unsigned32          hi;
} unsigned64_t;
```

```
/*
**  Add two unsigned 64-bit long integers.
*/
#define ADD_64b_2_64b(A, B, sum) \
    { \
        if (!(((A)->lo & 0x80000000UL) ^ ((B)->lo & 0x80000000UL))) { \
            if (((A)->lo&0x80000000UL)) { \
                (sum)->lo = (A)->lo + (B)->lo; \
                (sum)->hi = (A)->hi + (B)->hi + 1; \
            } \
            else { \
                (sum)->lo  = (A)->lo + (B)->lo; \
                (sum)->hi = (A)->hi + (B)->hi; \
            } \
        } \
        else { \
            (sum)->lo = (A)->lo + (B)->lo; \
            (sum)->hi = (A)->hi + (B)->hi; \
            if (!((sum)->lo&0x80000000UL)) (sum)->hi++; \
        } \
    }

/*
**  Add a 16-bit unsigned integer to a 64-bit unsigned integer.
*/
#define ADD_16b_2_64b(A, B, sum) \
    { \
        (sum)->hi = (B)->hi; \
        if ((B)->lo & 0x80000000UL) { \
            (sum)->lo = (*A) + (B)->lo; \
            if (!((sum)->lo & 0x80000000UL)) (sum)->hi++; \
        } \
        else \
            (sum)->lo = (*A) + (B)->lo; \
    }
```

```
/*
** Global variables.
*/
static unsigned64_t   time_last;
static unsigned16    clock_seq;

static void
mult32(unsigned32 u, unsigned32 v, unsigned64_t *result)
{
    /* Following the notation in Knuth, Vol. 2. */
    unsigned32 uuid1, uuid2, v1, v2, temp;

    uuid1 = u >> 16;
    uuid2 = u & 0xFFFF;
    v1 = v >> 16;
    v2 = v & 0xFFFF;
    temp = uuid2 * v2;
    result->lo = temp & 0xFFFF;
    temp = uuid1 * v2 + (temp >> 16);
    result->hi = temp >> 16;
    temp = uuid2 * v1 + (temp & 0xFFFF);
    result->lo += (temp & 0xFFFF) << 16;
    result->hi += uuid1 * v1 + (temp >> 16);
}

static void
get_system_time(unsigned64_t *uuid_time)
{
    struct timeval tp;
    unsigned64_t utc, usecs, os_basetime_diff;

    gettimeofday(&tp, (struct timezone *)0);
    mult32((long)tp.tv_sec, 10000000, &utc);
    mult32((long)tp.tv_usec, 10, &usecs);
    ADD_64b_2_64b(&usecs, &utc, &utc);

    /* Offset between UUID formatted times and Unix formatted times.
     * UUID UTC base time is October 15, 1582.
     * Unix base time is January 1, 1970. */
    os_basetime_diff.lo = 0x13814000;
    os_basetime_diff.hi = 0x01B21DD2;
    ADD_64b_2_64b(&utc, &os_basetime_diff, uuid_time);
}
```

```
/*
** See "The Multiple Prime Random Number Generator" by Alexander
** Hass pp. 368-381, ACM Transactions on Mathematical Software,
** 12/87.
*/
static unsigned32 rand_m;
static unsigned32 rand_ia;
static unsigned32 rand_ib;
static unsigned32 rand_irand;

static void
true_random_init(void)
{
    unsigned64_t t;
    unsigned16 seed;

 /* Generating our 'seed' value Start with the current time, but,
  * since the resolution of clocks is system hardware dependent and
  * most likely coarser than our resolution (10 usec) we 'mixup' the
  * bits by xor'ing all the bits together.  This will have the effect
  * of involving all of the bits in the determination of the seed
  * value while remaining system independent.  Then for good measure
  * to ensure a unique seed when there are multiple processes
  * creating UUIDs on a system, we add in the PID.
  */
    rand_m = 971;
    rand_ia = 11113;
    rand_ib = 104322;
    rand_irand = 4181;
    get_system_time(&t);
    seed  =  t.lo        & 0xFFFF;
    seed ^= (t.lo >> 16) & 0xFFFF;
    seed ^=  t.hi        & 0xFFFF;
    seed ^= (t.hi >> 16) & 0xFFFF;
    rand_irand += seed + getpid();
}

static unsigned16
true_random(void)
{
    if ((rand_m += 7) >= 9973)
        rand_m -= 9871;
    if ((rand_ia += 1907) >= 99991)
        rand_ia -= 89989;
    if ((rand_ib += 73939) >= 224729)
        rand_ib -= 96233;
    rand_irand = (rand_irand * rand_m) + rand_ia + rand_ib;
    return (rand_irand >> 16) ^ (rand_irand & RAND_MASK);
}
```

```
/*
**  Startup initialization routine for the UUID module.
*/
void
uuid_init(void)
{
    true_random_init();
    get_system_time(&time_last);
#ifdef NONVOLATILE_CLOCK
    clock_seq = read_clock();
#else
    clock_seq = true_random();
#endif
}

static int
time_cmp(unsigned64_t *time1, unsigned64_t *time2)
{
    if (time1->hi < time2->hi) return -1;
    if (time1->hi > time2->hi) return 1;
    if (time1->lo < time2->lo) return -1;
    if (time1->lo > time2->lo) return 1;
    return 0;
}

static void new_clock_seq(void)
{
    clock_seq = (clock_seq + 1) % (CLOCK_SEQ_LAST + 1);
    if (clock_seq == 0) clock_seq = 1;
#ifdef NONVOLATILE_CLOCK
    write_clock(clock_seq);
#endif
}
```

```
void uuid_create(uuid_t *uuid)
{
    static unsigned64_t     time_now;
    static unsigned16       time_adjust;
    byte                    eaddr[6];
    int                     got_no_time = 0;

    get_ieee_node_identifier(&eaddr);      /* TO BE PROVIDED */

    do {
        get_system_time(&time_now);
        switch (time_cmp(&time_now, &time_last)) {
        case -1:
            /* Time went backwards. */
            new_clock_seq();
            time_adjust = 0;
            break;
        case 1:
            time_adjust = 0;
            break;
        default:
            if (time_adjust == 0x7FFF)
                /* We're going too fast for our clock; spin. */
                got_no_time = 1;
            else
                time_adjust++;
            break;
        }
    } while (got_no_time);

    time_last.lo = time_now.lo;
    time_last.hi = time_now.hi;

    if (time_adjust != 0) {
        ADD_16b_2_64b(&time_adjust, &time_now, &time_now);
    }

    /* Construct a uuid with the information we've gathered
     * plus a few constants. */
    uuid->time_low = time_now.lo;
    uuid->time_mid = time_now.hi & 0x0000FFFF;
    uuid->time_hi_and_version = (time_now.hi & 0x0FFF0000) >> 16;
    uuid->time_hi_and_version |= (1 << 12);
    uuid->clock_seq_low = clock_seq & 0xFF;
    uuid->clock_seq_hi_and_reserved = (clock_seq & 0x3F00) >> 8;
    uuid->clock_seq_hi_and_reserved |= 0x80;
    memcpy(uuid->node, &eaddr, sizeof uuid->node);
}
```

Attachment L: DHCP Options For Host System Characteristics

INTERNET DRAFT Mike Henry
Eric Dittert
Intel Corp.
March 26, 1997

DHCP Options For Host System Characteristics

<draft-dittert-host-sys-char-01.txt>

Status of this Memo

This document is an Internet Draft. Internet Drafts are working documents of the Internet Engineering Task Force (IETF), its Areas, and its Working Groups. Note that other groups may also distribute working documents as Internet Drafts.

Internet Drafts are draft documents valid for a maximum of six months. Internet Drafts may be updated, replaced, or obsoleted by other documents at any time. It is not appropriate to use Internet Drafts as reference material or to cite them other than as a "working draft" or "work in progress."

Please check the "1id-abstracts.txt" listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), nic.nordu.net (Europe), munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or ftp.isi.edu (US West Coast).

Notice

Abstract

The interoperability of configuration services based on the Dynamic Host Configuration Protocol (DHCP) [1] in an environment of heterogeneous clients depends on clients accurately identifying themselves and their relevant characteristics to configuration servers. The class identifier provided through DHCP option 60 [2] helps in this regard, but such identifiers essentially only enable clients and servers that are "good friends" to find each other. This draft proposes the definition of two options that convey particular, generally useful information about the client system. This enables all servers to recognize this information, and is a step toward a richer form of interoperability for configuration services.

The proposed options are

* Client System Architecture
* Client Network Device Interface

This draft also proposes a new type of client identifier based on generated UUID/GUIDs to be used in conjunction with the DHCP client identifier option (61).

1.0 Introduction

The use of DHCP to provide clients with configuration information in general, and boot images in particular can be complicated by several circumstances. Among these are

1) clients in the same service domain with different system architectures or hardware configurations

2) clients in the same service domain for which different software configurations are desired

3) the desire to have clients and servers provided by different vendors successfully interact

(By “clients in the same service domain” we mean clients, requests from which can reach the same server.) A key element in enabling the successful use of DHCP in such circumstances is the provision of mechanisms by which clients can accurately identify themselves and their relevant characteristics to a server.

For identifying characteristics of the client that are relevant to the selection of a boot image, the currently available mechanisms are the DHCP class identifier option (code 60) and the DHCP vendor specific information option (code 43). By definition, the vendor specific information option does not address the problem of enabling interoperability of clients and servers provided by different vendors. Information conveyed by the class identifier option could enable interoperability, provided that a sufficiently specific and complete set of class identifiers were defined and agreed to.

We suggest using an alternate approach, in which new, specific options are used to convey the characteristics of the client that determine which boot image(s) could run on the client, and the class identifier is used as a (site-specific) designation of the desired software configuration for the client. Section 2 defines two new options that are useful for conveying the client’s hardware configuration.

For identifying the client as a unique entity, the currently available mechanisms is the DHCP client identifier option (code 61) [2]. Section 3 of this draft defines for use in this option an identifier type based on generated GUIDs - identifiers that are guaranteed to be, or are very, very likely to be unique across time and all clients.

2.0 Client Characteristics Options

The options defined in this section provide the server with explicit knowledge about the client system that is generally useful in selecting an executable that the client can use as a boot image.

2.1 Client System Architecture Option

DHCP clients SHOULD include this option in DHCPDISCOVER and DHCPREQUEST messages. Doing so provides the server with explicit knowledge of the client's system architecture.

DHCP servers that use this option SHOULD include the option in responses that contain a bootfile name. If included, the value of the option MUST denote a system architecture for which the bootfile named is valid. DHCP servers MUST NOT include this option in responses that do not contain a bootfile name.

The format for this option is as follows:

```
 Code  Len   System Arch Code
+-----+-----+-----+-----+
| TBD |  2  | s1  | s2  |
+-----+-----+-----+-----+
```

The currently defined types and their codes are

```
System Architecture                Code
-------------------                ----
Intel Architecture PC                1
NEC PC-9800                          2
```

2.2 Client Network Device Interface Option

DHCP clients SHOULD include this option in DHCPDISCOVER and DHCPREQUEST messages. Doing so provides the server with explicit knowledge of the client's network device.

DHCP servers that use this option SHOULD include the option in responses that contain a bootfile name. If included, the value of the option MUST denote a network device for which the bootfile named is valid. DHCP servers MUST NOT include this option in responses that do not contain a bootfile name.

Three types of network device specifications are defined for use with this option:

* devices that support the Universal Network Driver Interface (UNDI), as described in the Net PC design guidelines [3]
* Plug-and-Play devices [4]
* PCI devices [5]

Each devices that supports (UNDI) SHOULD be specified as an UNDI device, regardless of whether it is also a Plug-and-Play device or a PCI device. To specify an UNDI device, the option contains a type code of 1 and the major and minor UNDI version numbers:

```
 Code  Len   Type  Major Minor
+-----+-----+-----+-----+-----+
| TBD |  3  |  1  | m1  | m2  |
+-----+-----+-----+-----+-----+
```

To specify a PCI network device, a type code of 2 is used, and the vendor ID, device ID, class code, and revision are included:

```
 Code  Len   Type  Vendor ID   Device ID   Class code        Rev
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| TBD |  9  |  2  | v1  | v2  | d3  | d4  | c1  | c2  | c3  | r1  |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
```

To specify a Plug-and-Play network device, a type code of 3 is used, and the EISA device ID and the class code are included:

```
 Code  Len   Type  EISA device ID          Class code
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| TBD |  8  |  3  | e1  | e2  | e3  | e4  | c1  | c2  | c3  |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
```

3.0 UUID/GUID-based Client Identifiers

Whenever a client identifier option is included in a DHCP message, it MAY contain an identifier in UUID/GUID format. A client identifier option containing a type code of <TBD> MUST contain a 128-bit GUID as follows:

```
 Code   Len    Type   Client GUID
+------+------+------+------+------+------+
|  61  |  17  |  t1  |  g1  |  g2  |  ... |
+------+------+------+------+------+------+
```

The format of the GUID MUST be as specified in the design guidelines for Net PCs [3].

4.0 References

[1] Droms, R. “Dynamic Host Configuration Protocol”, RFC 1531

[2] Alexander,S. and Droms, R., “DHCP Options and BOOTP Vendor Extension” RFC 1533.

[3] Design Guidelines for a Net PC, reference to be provided

[4] Plug-and-Play specification, reference to be provided

[5] PCI specification, reference to be provided

5.0 Authors’ Addresses

Mike Henry
Intel Corporation, MS JF3-408
5200 NE Elam Young Pkwy
Hillsboro, OR 97124

Phone: (503) 264-9689
Email: Mike_Henry@ccm.jf.intel.com

Eric Dittert
Intel Corporation, MS JF3-206
5200 NE Elam Young Pkwy
Hillsboro, OR 97124

Phone: (503) 264-8461
Email: Eric_Dittert@ccm.jf.intel.com

APPENDIX F

PC 98 Master Checklist

If a recommended feature is implemented, it must meet the PC 98 requirements for that feature as defined in this document.

Checklist for Basic PC 98

Consumer PC 98	**Office PC 98**	**Entertainment PC 98**
1. System performance meets PC 98 minimum requirements		
Required	*Required*	*Required*
2. System design meets ACPI 1.0 specification and PC 98 requirements		
Required	*Required*	*Required*
3. Hardware design supports OnNow initiative		
Required	*Required*	*Required*
4. BIOS meets PC 98 requirements for OnNow support		
Required	*Required*	*Required*
5. BIOS meets PC 98 requirements for boot support		
Required	*Required*	*Required*
6. All expansion slots in the system are accessible for users to insert cards		
Required	*Required*	*Required*
7. Audible noise meets PC 98 requirements		
Required	*Required*	*Required*
8. System and component design practices follow accessibility guidelines		
Recommended	*Recommended*	*Recommended*
9. Internal system modification capabilities are not accessible to end users		
Recommended	*Recommended*	*Recommended*
10. System design provides physical security		
Recommended	*Recommended*	*Recommended*
11. Each device and driver meets PC 98 device requirements		
Required	*Required*	*Required*
12. Each bus and device meets Plug and Play specifications		
Required	*Required*	*Required*

Consumer PC 98	Office PC 98	Entertainment PC 98
13. Unique Plug and Play device ID provided for each system device and add-on device		
Required	*Required*	*Required*
14. Option ROMs meet Plug and Play requirements		
Required	*Required*	*Required*
15. "PNP" vendor code used only to define a legacy device's CompatibleID		
Required	*Required*	*Required*
16. Device driver and installation meet PC 98 requirements		
Required	*Required*	*Required*
17. Minimal user interaction needed to install and configure devices		
Required	*Required*	*Required*
18. Connections use icons plus keyed or shrouded connectors		
Required	*Required*	*Required*
19. Hot-plugging capabilities for buses and devices meet PC 98 requirements		
Required	*Required*	*Required*
20. Device Bay-capable bay and peripherals meet Device Bay specification		
Required	*Required*	*Required*
21. Multifunction add-on devices meet PC 98 device requirements for each device		
Required	*Required*	*Required*
22. All devices support correct 16-bit decoding for I/O port addresses		
Required	*Required*	*Required*
23. System-board devices use ISA-compatible addresses		
Required	*Required*	*Required*
24. Each bus meets written specifications and PC 98 requirements		
Required	*Required*	*Required*
25. System includes USB with one USB port, minimum		
Required	*Required*	*Required, with 2 USB ports*
26. System includes support for other high-speed expansion capabilities		
Recommended	*Recommended*	*Required, with 2 IEEE 1394 ports*
27. If present, PCI bus meets PCI 2.1 or higher, plus PC 98 requirements		
Required	*Required*	*Required*
28. System does not include ISA expansion devices		
Required	*Required*	*Required, with no ISA slots*
29. System includes keyboard connection and keyboard		
Required	*Required*	*Required; USB or wireless*
30. System includes pointing-device connection and pointing device		
Required	*Required*	*Required; USB or wireless*
31. System includes connection for external parallel devices		
Required	*Required*	*Required*
32. System includes connection for external serial devices		
Required	*Required*	*Required*

Consumer PC 98	Office PC 98	Entertainment PC 98
33. System includes wireless capabilities		
Recommended	*Recommended*	*Recommended*
34. System includes support for operating system installation		
Required	*Required*	*Required*
35. System audio support meets PC 98 requirements		
Recommended	*Recommended*	*Required*
36. System includes communications device		
Required; modem	*Required; network adapter*	*Required; modem*
37. System includes smart card support		
Not applicable	*Recommended*	*Not applicable*
38. Graphics adapter meets PC 98 minimum requirements		
Required	*Required*	*Required*
39. Adapter supports television output if system does not include a large-screen monitor		
Recommended	*Recommended*	*Recommended*
40. Color monitor is DDC-compliant with unique EDID identifier		
Required	*Required*	*Required*
41. System meets PC 98 DVD-Video and MPEG-2 playback requirements		
Required	*Required with DVD-Video*	*Required*
42. System supports PC 98 analog video input and capture capabilities		
Recommended	*Recommended*	*Required*
43. System includes analog television tuner		
Recommended	*Recommended*	*Required*
44. System BIOS and option ROMs support Int 13h Extensions		
Required	*Required*	*Required*
45. Host controller for storage device meets PC 98 requirements		
Required	*Required*	*Required*
46. Host controllers and devices support bus mastering		
Required	*Required*	*Required*
47. Hard drive meets PC 98 requirements		
Required	*Required*	*Required*
48. Removable media support media status notification		
Required	*Required*	*Required*
49. Floppy disk capabilities are provided using an expansion card or external bus		
Recommended	*Recommended*	*Recommended*
50. System supports WHIIG		
Not applicable	*Required*	*Not applicable*
51. System includes driver support for WMI		
Not applicable	*Required*	*Not applicable*
52. Management information service provider enabled by default		
Not applicable	*Required*	*Not applicable*
53. Expansion devices can be remotely managed		
Not applicable	*Required*	*Not applicable*

Checklist for Workstation PC 98

1. Workstation meets all requirements for Office PC 98
Required

2. Workstation meets requirements for optimal performance
Required

3. Workstation supports multiple processors
Recommended

4. Workstation RAM can be expanded
Recommended

5 Workstation system memory includes ECC memory protection
Required

6. Workstation includes APIC support
Required

7. Workstation includes high-performance components
Recommended

8. Workstation supports 64-bit I/O bus architecture
Recommended

9. Workstation does not include ISA expansion slots
Recommended

10. Graphics subsystem supports workstation performance demands
Required

11. Workstation meets PC 98 DVD-Video and MPEG-2 playback requirements
Required, with DVD-Video

12. Storage components rely on SCSI controller
Recommended

13. Workstation includes multiple hard drives
Recommended

Checklist for Entertainment PC 98

1. Entertainment PC 98 system includes two USB ports, with at least one easily accessible connector
Required

2. Entertainment PC 98 system includes two IEEE 1394 ports, with at least one easily accessible connector
Required

3. Entertainment PC 98 system does not include ISA expansion slots
Required

4. All Entertainment PC 98 input devices meet USB HID specifications
Required

5. Entertainment PC 98 includes remote-control pointing device
Recommended

6. All Entertainment PC 98 pointing devices support DirectInput and work simultaneously
Required

7. Entertainment PC 98 includes data/fax/voice modem that supports V.pcm
Required

8. Entertainment PC 98 includes DVD-ROM drive and DVD-Video playback
Required

9. Audio support meets PC 98 audio and Entertainment PC 98 requirements
Required

10. Graphics adapter uses AGP
Required

11. Entertainment PC 98 graphics subsystem includes PC 98 hardware acceleration for 2-D and 3-D graphics
Required

12. Entertainment PC 98 graphics subsystem includes support for television output if the system doesn't have a large-screen monitor
Recommended

13. Entertainment PC 98 includes large-screen DDC2B color entertainment monitor
Recommended

14. Entertainment PC 98 DVD-Video and MPEG-2 playback meet PC 98 requirements
Required

15. Entertainment PC 98 supports PC 98 analog video input and capture capabilities
Required

16. Entertainment PC 98 includes analog television tuner
Required

17. Entertainment PC 98 includes digital broadcast satellite subsystem
Recommended

18. Entertainment PC 98 includes DTV support
Required

Checklist for Mobile PC 98

1. Mobile PC performance meets PC 98 minimum requirements
Required

2. Mobile PC supports Smart Battery or ACPI-specified battery
Required

3. Expansion capabilities of mobile PC are accessible to users
Required

4. Mobile PC connections use icons plus keyed or shrouded connectors
Required

5. Mobile PC includes a USB port
Required

6. Mobile PC includes an IEEE 1394 port
Recommended

7. USB-connected device does not maintain fully on power state
Required

8. Mobile PC includes CardBus
Required

9. Mobile PC keyboard and pointing device meet PC 98 requirements
Required

10. Mobile PC includes wireless capabilities
Recommended

11. Mobile PC includes support for installing the operating system
Required

12. Mobile PC audio meets PC 98 audio requirements
Recommended

13. Mobile PC includes communications device
Recommended

14. Built-in display adapter meets PC 98 minimum requirements
Required

15. Mobile system supports hot pluggable devices and alternative server connections
Recommended

16. Mobile PC/docking station combination meets PC 98 requirements
Required

17. Docking station meets all Basic PC 98 requirements
Required

18. Docking station interface is supported using ACPI-defined mechanisms
Required

19. Mobile PC/docking station combination supports automatic resource assignment and dynamic disable capabilities
Required

20. Docking station supports warm docking
Required

21. Docking system supports fail-safe docking
Required

22. Port replicator supports automatic resource assignment and dynamic disable capabilities for replacement devices
Required

23. Port replicator supports warm docking
Required

24. Mini-notebook performance meets PC 98 minimum requirements
Required

Checklist for USB

1. USB included on PC 98 system
Required

2. All USB hardware complies with USB 1.0 specifications
Required

3. Connections use USB icon
Required

4. Devices and drivers support maximum flexibility of hardware interface options
Recommended

5. USB host controller meets either OpenHCI or UHCI specification
Required

6. USB host controller can wake the system
Required

7. System and devices comply with USB power management requirements
Required

8. USB devices meet requirements in related USB device class specification
Required

Checklist for IEEE 1394

1. Controllers and devices support IEEE 1394-1995 standards
Required

2. Controllers comply with OpenHCI for IEEE 1394
Required

3. OpenHCI controllers and devices support advances defined in IEEE 1394A
Required

4. Host supports peak data rate of 400 Mb/s, minimum
Required

5. Design avoids excessive currents resulting from ground-fault potential among devices
Recommended

6. Device command protocols conform to standard device class interfaces
Required

7. Devices support peak data rate of 400 Mb/s, minimum
Recommended

8. Devices requiring support for high-bandwidth data transfer use IEEE 1394
Recommended

9. Plug and Play devices demonstrate interoperability with other devices
Required

10. Topology faults do not cause the bus to fail
Required

11. Removable media devices support media status notification
Required

12. Devices that can initiate peer-to-peer communications also support remote programming
Required

13. Device provides a configuration ROM for unique device identification
Required

14. Device configuration ROM implements general ROM format
Required

15. Bus information block implemented at a base address offset of 0404h
Required

16. Configuration ROM provides globally unique device ID
Required

17. Root directory is located at a fixed address following the bus information block
Required

18. Configuration ROM includes a unit directory for each independent device function
Required

19. Each unit directory provides a valid Unit_Spec_Id and Unit_Sw_Version
Required

20. Each unit directory provides a pointer to a unit-dependent directory
Required

21. Vendor and model leaves support textual descriptor leaf format
Required

22. Unit-dependent directory provides a pointer to the unit's CSRs
Required

23. Device provides three connector ports
Recommended

24. Device uses standard 6-pin IEEE 1394 connector
Required

25. Self-powered devices propagate the power bus through each connector
Required

26. Only single-port leaf-node devices use 4-pin connectors
Required

27. Device connectors exhibit common speed and power characteristics
Required

28. Standard 400-Mb/s rated IEEE 1394 cable provided with devices
Required

29. Devices power their PHY at all times
Required

30. Devices report power source and cable power consumption in Self_id packet
Required

31. Devices implement link power control
Required

32. Device requiring power increments in excess of Link_on implements unit-power CSRs
Required

33. Devices that source cable power must report this capability
Required

34. IEEE 1394-enabled PC sources cable power
Required

35. Power source supplies a minimum of 20 volts at 15 watts
Recommended

36. Devices notify the power manager of power change requests
Required

37. Devices and controllers comply with Cable Power Distribution specification
Required

38. Devices and controllers comply with IEEE 1394 power specification
Required

Checklist for PCI

1. All components comply with PCI 2.1
Required

2. System does not contain ghost cards
Required

3. System uses standard method to close BAR windows on nonsubtractive decode PCI bridges
Required

4. System supports PCI docking through a bridge connector
Recommended

5. PCI chip sets support Ultra DMA/33
Required

6. System-board bus complies with PCI 2.1
Required

7. Bus master privileges are supported for all connectors
Required

8. ISA Write Data Port address is propagated to the ISA bus at power up
Required

9. Functions in a multifunction PCI device do not share writable PCI Configuration Space bits
Required

10. Devices use PCI 2.1 Configuration Space register for Plug and Play device ID
Required

11. Device IDs include PCI 2.1 Subsystem IDs
Required

12. Configuration Space is correctly populated
Required

13. Interrupt routing supported using ACPI
Required

14. BIOS does not configure I/O systems to share PCI interrupts
Recommended

15. BIOS configures boot device IRQ and writes to the interrupt line register
Required

16. Hot swapping for any PCI device uses ACPI-based methods
Required

17. All PCI components comply with PCI Bus Power Management Interface specification
Required

Checklist for IDE and ATAPI

1. Controller complies with ATA-2 specification
Required

2. Bootable IDE controller supports El Torito No Emulation mode
Required

3. System BIOS and option ROMs support Int 13h Extensions
Required

4. Controller and peripherals support media status notification
Required

5. Dual IDE adapters use single FIFO with asynchronous access or dual FIFOs and channels
Required

6. System BIOS and devices support LBA
Required

7. Controller and peripherals support PCI IDE bus mastering
Required

8. Controller and peripheral connections include Pin 1 cable designation with keyed and shrouded connectors
Required

9. Peripherals comply with SFF 8020i, Version 2.5 or higher
Required

10. BIOS enumeration of all ATAPI devices complies with SFF 8020i, Version 2.5 or higher
Required

11. Devices support ATAPI RESET command
Recommended

12. IDE/ATAPI controllers and devices support Ultra DMA/33
Required

13. Operating system recognizes the boot drive in a multiple-drive system
Required

14. Each device has a Plug and Play device ID
Required

15. Dynamic resource configuration supported for all devices
Required

16. Resource configuration meets bus requirements
Required

17. ISA address ranges 3F7h and 377h are not claimed by IDE controllers
Required

18. Device supports ATA STANDBY command
Required

19. Bus and device meet PC 98 power management requirements
Required

Checklist for SCSI

1. Host controller supports bus mastering
Required

2. Option ROMs support Int 13h Extensions
Required

3. Option ROMs support virtual DMA services
Required

4. Bus type is clearly indicated on connectors for all adapters, peripherals, and terminators
Required

5. Differential devices support DIFFSENS as defined in SCSI-3
Required

6. Automatic termination circuit meets SCSI-3 specification
Required

7. SCSI terminator built onto internal cables meets SCSI-3 specification
Required

8. Terminator power is supplied to the SCSI bus, with over-current protection
Required

9. High-density external connector meets SCSI-2 specification
Required

10. Internal terminator is close as possible to the last peripheral on the cable
Recommended

11. SCSI bus parity signal meets SCSI-2 specification
Required

12. Cables meet SCSI-3 Clause 6 requirements
Required

13. User cannot incorrectly plug in cables for internal connections
Required

14. Internal SCSI peripherals do not terminate the SCSI bus
Recommended

15. External connectors use automatic termination or an accessible on-board termination switch
Required

16. High-density, shielded device connector meets SCSI-2 specification
Recommended

17. Removable media includes media status notification support
Recommended

18. All components comply with Plug and Play SCSI specifications
Required

19. Each device has a Plug and Play device identifier
Required

20. Automatic resource assignment and dynamic disable capabilities are supported for all devices
Required

21. Bus and device meet PC 98 power requirements
Required

22. Hardware supports the STOP/START UNIT command as defined in SCSI-2
Required

23. STOP/START UNIT command can be used to decrease power consumption
Recommended

Checklist for PC Card

1. All devices comply with the PC Card standards
Required

2. System and ZV-compatible PC Card 16 cards comply with ZV standard definitions
Required

3. Controller supports industry-standard ExCA register set
Required

4. System maintains mapping of IRQ Routing Register bits to system interrupt vectors
Required

5. IRQ connections can be determined by using the 0805 register
Required

6. CardBus controllers support both ISA and PCI interrupts
Required

7. System supports industry-standard definition for CardBus bridges
Required

8. BIOS initializes CardBus controller in 82365-compatible mode and supports backward compatibility
Recommended

9. CardBus controllers do not share writable PCI Configuration Space bits
Required

10. Each PC Card 16 memory window in CardBus controller has it own page register
Required

11. Card supports required I/O card tuples
Required

12. Configuration table entry tuples listed in priority order
Required

13. Card specifies maximum configuration options
Required

14. Configuration space meets Common Silicon Guidelines
Required

15. RESERVED fields comply with PCI 2.1
Required

16. CardBus card implements required and recommended tuples
Required

17. Socket controller complies with device class power management reference specification
Required

18. PC Card 16 cards implement power-related events using ReqAttn bit and #STSCHG mechanism
Required

19. CardBus controllers and cards implement power management specifications
Required

20. No user intervention required for correctly installing devices
Required

21. Device is immediately functional without restarting the system
Required

22. ZV-compatible PC Card driver uses DirectDraw LVE
Required

23. PC Card 16 card driver supports sharing of level-mode interrupts
Required

Checklist for I/O Ports and Devices

Consumer PC 98	Office PC 98	Entertainment PC 98
1. System includes connection for external serial devices		
Required	*Required*	*Required*
2. System includes connection for external parallel devices		
Required	*Required*	*Required*
3. System includes external connection for keyboard		
Required	*Required*	*Required*
4. System includes external connection for pointing device		
Required	*Required*	*Required*
5. System includes USB game pad or joystick		
Recommended	*Recommended*	*Recommended; wireless*
6. System includes built-in wireless capabilities		
Recommended	*Recommended*	*Recommended*
7. Devices use USB or external bus connections rather than legacy serial or parallel ports		
Required	*Recommended*	*Required*
8. All devices meet PC 98 general device requirements		
Required	*Required*	*Required*
9. Serial port meets device class specifications for its bus		
Required		

Consumer PC 98	Office PC 98	Entertainment PC 98

10. Legacy serial port is implemented as 16550A UART or equivalent and supports 115.2K baud
Required

11. Legacy serial port supports flexible resource configuration and dynamic disable capabilities
Required

12. Conflict resolution for legacy serial port ensures availability of at least one serial port
Required

13. Parallel port meets device class specifications for its bus
Required

14. Flexible resource configuration supported for each parallel port
Required

15. EPP support does not use restricted I/O addresses
Required

16. Compatibility, nibble mode, and ECP protocols meet IEEE 1284-1994 specifications
Required

17. Port connectors meet IEEE 1284-I specifications, minimum
Required

18. IEEE 1284 peripherals have Plug and Play device IDs
Required

19. Device identification string provides CompatibleID key
Recommended

20. Pointing-device connection meets requirements for its bus class
Required

21. Remote control provides PC 98 minimum support
Recommended

22. Keyboard connection meets requirements for its bus class
Required

23. No interference occurs between multiple keyboards
Required

24. Keyboard includes Windows and Application logo keys
Recommended

25. Device meets USB HID class specification requirements
Required

26. IR device uses NDIS 5.0 miniport driver
Required

27. IR device meets IrDA specifications
Required

28. IR device meets IrDA Control IR specification
Required

Consumer PC 98	Office PC 98	Entertainment PC 98
29. IR device meets PC 98 bus and port specifications		
Required		
30. IR device meets USB guidelines for interfacing with IrDA and Control IR devices		
Required		
31. IR device supports flexible resource configuration and dynamic disable capabilities		
Required		
32. System supports standard input speeds of 4 Mb/s		
Recommended		
33. System differentiates command streams if transceiver includes legacy consumer IR support		
Required		
34. Each device has a unique Plug and Play device ID		
Required		
35. Automatic resource assignment and dynamic disable capabilities are supported		
Required		
36. Each device complies with its device class power management reference specification		
Required		
37. Device supports wake-up events		
Required for wireless input;		
optional for other devices		
38. Device drivers and installation meet PC 98 requirements		
Required		

Checklist for Graphics Adapters

Consumer PC 98	Office PC 98	Entertainment PC 98
1. Graphics adapter uses PCI, AGP, or another high-speed bus		
Required	*Required*	*AGP required*
2. System uses WC with higher-performance processors		
Required	*Required*	*Required*
3. Primary graphics adapter works normally with default VGA mode driver		
Required	*Required*	*Required*
4. Adapter and driver support multiple adapters and multiple monitors		
Required	*Required*	*Required*
5. Adapter supports television output if system does not include large-screen monitor		
Recommended	*Recommended*	*Required*

Consumer PC 98	Office PC 98	Entertainment PC 98
6. Adapter meets PC 98 general device requirements		
Required		
7. Screen resolution and local memory capacity meet PC 98 minimum requirements		
Required	*Required*	*Required*
8. Adapter meets VESA specifications for ergonomic timing rates		
Required		
9. All supported color depths are enumerated		
Required		
10. Graphics operations use relocatable registers only		
Required		
11. Adapter supports downloadable RAMDAC entries for image color matching		
Required		
12. Adapter supports DDC monitor detection		
Required		
13. Adapter supports video overlay surface with scaling		
Required	*Required*	*Required*
14. Hardware supports VGA destination color keying for video rectangle		
Required	*Required with DVD-Video*	*Required*
15. Video port meets PC 98 specifications if present on graphics adapter		
Required	*Required*	*Required*
16. Adapter supports MPEG-2 motion compensation acceleration		
Recommended		
17. Extended resources can be dynamically relocated after system boot		
Required		
18. VGA resources can be disabled by software		
Required		
19. Frame buffer can be accessed directly by applications		
Required	*Required*	*Required*
20. Adapter and driver support linear-mapped, low-resolution modes		
Required	*Required*	*Required*
21. Adapter supports transparent blter		
Required	*Required*	*Required*
22. Hardware supports double buffering with no tearing		
Required	*Required*	*Required*
23. Hardware supports programmable blter stride		
Required	*Recommended*	*Required*
24. Hardware supports RGB rasterization		
Required	*Required*	*Required*
25. Hardware supports multi-texturing		
Recommended	*Recommended*	*Required*

Consumer PC 98	Office PC 98	Entertainment PC 98
26. Hardware supports texture formats		
Required	*Required*	*Required*
27. Hardware complies with texture size limitations		
Required	*Recommended*	*Required*
28. Hardware supports destination RGB alpha blending		
Recommended	*Recommended*	*Recommended*
29. Hardware supports Z comparison modes and Direct3D-compatible formats		
Recommended	*Recommended*	*Required*
30. Hardware meets PC 98 3-D accelerator performance requirements		
Recommended	*Recommended*	*Required*
31. Adapter supports both NTSC and PAL output		
Recommended		
32. Default boot mode supports appropriate locale		
Required	*Required*	*Required*
33. Adapter supports underscan scaling		
Required	*Recommended*	*Required*
34. Adapter supports flicker filter		
Required		
35. Adapter provides proper termination		
Required		
36. Adapter supports RCA-style composite video and S-Video connectors		
Recommended	*Recommended*	*Required*
37. Adapter supports both VGA and television output		
Required		
38. Software supports positioning		
Required	*Recommended*	*Required*
39. Software supports detection of television connection		
Required	*Recommended*	*Required*
40. Each device has a Plug and Play device ID		
Required		
41. System supports conflict resolution, VGA compatibility, and extended registers		
Required		
42. Chips support linear packed-pixel frame buffer, relocatable above 16 MB		
Required		
43. Option ROM supports DDC2B		
Required		
44. BIOS setup utility provides option to force use of system-board graphics		
Recommended		
45. BIOS supports large frame buffers for graphics adapters		
Required		

Consumer PC 98	Office PC 98	Entertainment PC 98
46. AGP meets PC 98 implementation guidelines		
Required		
47. PCI graphics device supports IRQ and correctly populates PCI BARs		
Required	*Recommended*	*Required*
48. PCI system-board graphics device is not hidden from Plug and Play enumeration		
Required		
49. Graphics adapter complies with device class power management reference specification		
Required		
50. Graphics adapter complies with VBE/Core 2.0 extensions for power management		
Required		
51. Device drivers and installation meet PC 98 requirements		
Required		
52. Driver does not bypass any Microsoft-provided system components		
Required		
53. Applications provided with device meet Win32 requirements		
Required		
54. Driver supports dynamic color bit-depth change		
Required		

Checklist for Video and Broadcast Components

Consumer PC 98	Office PC 98	Entertainment PC 98
1. System meets PC 98 requirements for DVD-Video and MPEG-2 playback		
Required	*Required with DVD-Video*	*Required*
2. System supports PC 98 analog video input and capture capabilities		
Recommended	*Recommended*	*Required*
3. System includes analog television tuner		
Recommended	*Recommended*	*Required*
4. System includes digital broadcast or satellite subsystem		
Recommended	*Recommended*	*Recommended*
5. System includes DTV support		
Recommended	*Recommended*	*Required (U.S. only)*
6. Video input, capture, and broadcast device support is based on DirectX foundation class and WDM Stream class		
Required	*Required*	*Required*
7. Hardware MPEG-2 decoder uses video port for video data		
Required	*Required*	*Required*
8. PCI-based tuners and decoders support bus mastering with scatter/gather DMA		
Required		

Consumer PC 98	Office PC 98	Entertainment PC 98
9. Background tasks do not interfere with MPEG-2 playback		
Required	*Recommended*	*Required*
10. All components meet PC 98 general device requirements		
Required		
11. System warns users if it cannot play DVD movies		
Required	*Required if no DVD-Video*	*Required*
12. MPEG-2 playback meets PC 98 requirements		
Required	*Required with DVD-Video*	*Required*
13. Retail adapters with hardware MPEG-2 decoders enable a standard video port connection to the graphics adapter		
Required		
14. MPEG-2 decoder supports pull-down algorithm		
Recommended		
15. DVD decoder driver correctly handles media types, time discontinuity, and decode-rate adjustment		
Required		
16. DVD decoder supports subpicture compositing and closed captioning		
Required		
17. Subpicture decoder correctly handles subpicture properties and other functions		
Required		
18. System supports seamless DVD-Video 1.0 navigation		
Required		
19. System provides a licensed CSS copyright protection scheme		
Required		
20. Video input or capture device supports capture of NTSC/PAL picture quality		
Required		
21. Analog video capture device outputs video data rate of 3.7 MB per second, minimum		
Required		
22. Video input or capture device supports time-code reading		
Recommended		
23. Digital video camera uses external bus support		
Required		
24. Television tuner supports PC 98 audio and video performance		
Required		
25. Television tuner includes stereo tuner and supports SAP		
Recommended	*Recommended*	*Required*
26. VBI capture oversamples VBI data at least four times		
Required		
27. VBI capture detects validity of scan-line data		
Required		

Consumer PC 98	Office PC 98	Entertainment PC 98

28. VBI capture makes VBI data available to the CPU for processing
Required

29. Digital broadcast card can receive all video, audio, data, and other streams
Required

30. Digital broadcast card can receive full bandwidth from each frequency
Required

31. Digital broadcast card can receive a minimum of eight simultaneous streams
Required

32. System includes multiple digital broadcast tuner cards
Recommended

33. Digital broadcast card provides support for legacy conditional access
Required

34. Digital broadcast card provides signal quality and other diagnostic information
Required

35. Digital broadcast card supports general-purpose data cryptography
Recommended

36. Digital broadcast card supports substream filtering
Required

37. ATSC DTV tuner is fully implemented
Required

38. Stream splitting is supported using DirectShow filters
Recommended

39. MPEG-2 decoder and video port support ATSC DTV standard
Required

40. Each device has a Plug and Play device ID
Required

41. Conflict resolution and dynamic disable capabilities are supported
Required

42. Dependent video device is not independently enumerated
Required

43. Device drivers and installation meet PC 98 requirements
Required

44. Software drivers are installed during hardware driver installation
Required

45. Applications provided with device meet Win32 requirements
Required

46. NDIS 5.0 driver provided for digital broadcast receiver
Required

Checklist for Monitors

Consumer PC 98	Office PC 98	Entertainment PC 98
1. Color monitor is DDC2B-compliant with unique EDID identifier		
Required	*Required*	*Required*
2. Monitor supports ICC color matching		
Required		
3. Monitor meets all PC 98 general device and driver requirements		
Required		
4. Monitor meets minimum graphics resolution, based on monitor size		
Required		
5. Monitor supports ergonomic timing standards		
Required		
6. Large-screen monitor is 20 inches or larger, if included with an Entertainment PC system		
Required		
7. Large-screen monitor is 16:9, if included with PC system		
Recommended		
8. Entertainment monitor supports 800 × 600 at 60 Hz refresh rate		
Required		
9. Entertainment monitor's host control is DDC2B-compliant, with digitally controlled geometry		
Recommended		
10. External monitor meets DDC2B and EDID standards		
Required		
11. Monitor complies with device class power management reference specification		
Required		

Checklist for Audio Components

Consumer PC 98	Office PC 98	Entertainment PC 98
1. PC system includes PC 98 audio capabilities		
Recommended	*Recommended*	*Required*
2. Audio device does not connect to ISA bus		
Required	*Required*	*Required*

3. Audio performance meets PC 98 requirements
Required

4. Audio system provides support for basic data formats
Required

5. Audio system reports sample position for stream synchronization
Required

6. Audio system provides sufficient externally accessible inputs and outputs
Required

7. Audio system connectors are labeled with icons as defined for PC 98
Required

8. Audio performance meets PC 98 advanced recommendations
Recommended

9. Audio system supports full-duplex operation at independent sampling rates
Recommended

10. Audio system provides hardware or software support for the Downloadable Samples specification
Recommended

11. Audio system supports AEC reference inputs
Recommended

12. Audio system provides hardware filtering of HRTF 3-D filters
Optional

13. CD, DVD, and broadcast audio playback meet PC 98 requirements
Required with DVD Video

14. Audio system provides consistent volume levels for different devices
Optional

15. Each device has a unique Plug and Play device ID
Required

16. Automatic resource assignment and dynamic disable capabilities are supported
Required

17. PCI device conforms to PCI 2.1 and additional PC 98 requirements
Required

18. PCI device supports initiator, target, and block transfer
Required

19. PCI audio components use a suitable configuration scheme if using ISA resources
Required

Consumer PC 98	Office PC 98	Entertainment PC 98

20. PCI device is digital ready
Required

21. Audio meets USB specification and USB audio device class specification
Required

22. Audio meets PC 98 requirements for IEEE 1394
Required

23. System and device comply with PCI bus power management specification
Required

24. Audio device complies with device class power management reference specification
Required

25. Device supports wake-up events
Optional

26. Device drivers and installation meet PC 98 requirements
Required

27. Audio meets PC 98 requirements for WDM driver support
Required

28. Applications provided with device meet Win32 requirements
Required

Checklist for Storage and Related Peripherals

Consumer PC 98	Office PC 98	Entertainment PC 98

1. Storage device and controller support bus master capabilities
Required

2. Removable media includes media status notification support
Required

3. Option ROMs support Int 13h Extensions
Required

4. Device meets PC 98 general device requirements
Required

5. Device meets PC 98 requirements for ports or buses
Required

6. Device Bay storage device meets PC 98 requirements
Required

7. IDE/ATAPI devices supported on IEEE 1394
Recommended

8. IDE/ATAPI devices and controllers support Ultra DMA/33
Required

Consumer PC 98	Office PC 98	Entertainment PC 98

9. USB-based mass storage device meets PC 98 requirements for USB
Required

10. System BIOS or option ROM supports El Torito No Emulation mode
Required

11. Floppy disk capabilities provided through expansion card or external bus
Recommended

12. IDE floppy drive complies with SFF 8070
Required

13. Legacy FDC built into system
Optional

14. Legacy FDC device meets resource configuration requirements
Required

15. System supports conflict resolution and dynamic disable capabilities for legacy FDC
Required

16. IDE hard drive is SMART-compliant and uses SMART IOCTL API
Required

17. IDE hard drive spin-up time supports OnNow capabilities
Recommended

18. CD-ROM drive provides 8x or higher performance
Required

19. CD-ROM drive is CD-Enhanced–compatible
Required

20. CD-ROM drive supports specified logical and physical CD formats
Required

21. IDE/ATAPI CD-ROM drive complies with SFF 8020i, Version 1.2
Required

22. CD-ROM drive supports multisession and compatibility forms of the READ_TOC command
Required

23. IDE/ATAPI CD changer meets SFF 8070 specification
Required

24. ATAPI rewritable device meets SFF 8070i specification
Required

25. DVD drive supports bus master DMA transfers
Required

26. DVD drive meets minimum compatibility requirements
Required

27. Device and driver support DVD command sets
Required

Consumer PC 98	Office PC 98	Entertainment PC 98
28. DVD device meets SFF 8090 specification		
Required		
29. DVD device uses high-speed expansion bus		
Required		
30. DVD drive supports UDF		
Required		
31. DVD device uses push-to-close design		
Recommended		
32. DVD device supports defect management		
Required		
33. DVD device supports copyright protection		
Required		
34. Each device has a Plug and Play device ID		
Required		
35. Conflict resolution and dynamic disable capabilities supported for all devices		
Required		
36. 3F7h and 377h are unclaimed by devices		
Required		
37. Physical security is provided for storage devices		
Recommended	*Recommended*	*Recommended*
38. Device and controller comply with device class power management reference specification		
Required		
39. Device supports wake-up events		
Optional		
40. Device drivers and installation meet PC 98 requirements		
Required		
41. Device and file system run in protected mode following installation		
Required		
42. Applications provided with the device meet Win32 requirements		
Required		
43. Driver for partitioned media supports all Windows and Windows NT partition types		
Required		
44. Driver for block-mode device supports extended BPBs		
Required		

Checklist for Modems

Consumer PC 98	Office PC 98	Entertainment PC 98
1. Modem device is provided with PC system		
Required	*Required, if no network adapter*	*PCM modem required, upgradable to V.pcm*
2. Modem supports TIA-602 Hayes-compatible command set		
Required		
3. Data modem supports 33.6 Kbps (V.34-1996) with V.42 and V.42bis protocol		
Required		
4. Data modem supporting speeds faster than 33.6 Kbps can be upgraded to V.pcm		
Required		
5. Fax modem supports 14.4 Kbps (V.17) with Class 1 (TIA-578-A) command set		
Required		
6. Data modem supports V. 80 for synchronous access		
Recommended		
7. Modem supports adaptive connection, V.25, V.8, and V.8bis call control signaling with V.25ter Annex A modem commands		
Recommended		
8. Modem supports delayed and blacklisted number clearing		
Recommended		
9. Modem supports TDD, meeting V.18-1996 with V.25ter AT commands		
Recommended		
10. PCM modem supports ITU-T V.pcm		
Required		
11. Modem controller meets PC 98 requirements		
Required		
12. Voice modem supports TIA-695 (AT+V)		
Recommended	*Recommended*	*Required*
13. Voice modem support includes PC 98 recommendations		
Recommended		
14. Voice modem supports local telset interfaces		
Recommended		
15. Voice modem supports simultaneous voice/data integration capabilities		
Recommended		
16. Voice modem supports speakerphone		
Recommended		
17. Voice modem supports full-duplex voice I/O		
Recommended		
18. Wireless support implemented for modems		
Recommended		

Consumer PC 98	Office PC 98	Entertainment PC 98

19. Digital cellular phone support is implemented for modems
Recommended

20. ISDN modem supports required command set
Required

21. ISDN modem supports auto-SPID detection algorithms and standard SPID format
Required

22. ISDN modem supports CHAP in firmware if B channels are not exposed
Required

23. ISDN modem exposes both B channels
Recommended

24. ISDN modem supports multilink PPP
Recommended

25. ISDN modem supports asynchronous-to-synchronous conversion
Required

26. ISDN modem uses high-speed port
Recommended

27. ISDN driver supports switch detection
Recommended

28. ISDN driver supports unattended installation, with limitations
Required

29. Each device has a unique Plug and Play device ID
Required

30. Each device has a compatible Plug and Play device ID
Required

31. Automatic resource assignment and dynamic disable capabilities are supported
Required

32. PCI modem meets PCI 2.1 requirements
Required

33. USB modem meets USB specifications
Required

34. Device Bay modem meets PC 98 requirements
Required

35. Device complies with device class power management reference specification
Required

36. Device supports wake-up events
Required

37. Device drivers and installation meet PC 98 requirements
Required

38. Driver supports Unimodem
Required

39. Applications provided with device meet Win32 requirements
Required

Checklist for Network Communications

Consumer PC 98	Office PC 98	Entertainment PC 98
1. PC system includes network adapter		
Recommended	*Required, if no modem*	*Recommended*
2. PC system includes internal or external ISDN device		
Recommended	*Recommended*	*Recommended*
3. PC system includes cable modem		
Optional	*Optional*	*Optional*
4. PC system includes ATM adapter		
Optional	*Optional*	*Optional*
5. PC system includes ADSL adapter		
Optional	*Optional*	*Optional*
6. PC system includes satellite or broadcast receiver with NDIS driver		
Recommended	*Recommended*	*Recommended*
7. Adapter uses NDIS 5.0 miniport driver		
Required		
8. Full-duplex adapter automatically detects and switches to duplex mode		
Required		
9. Adapter automatically senses presence of functional network		
Required		
10. Adapter automatically senses transceiver type		
Required		
11. Adapter supports quadword buffer alignment for receive and byte buffer alignment for send		
Required		
12. Adapter communicates with driver across any bridge		
Required		
13. Adapter supports filtering for 32 multicast addresses, at minimum		
Required		
14. Adapter is compatible with remote new system setup capabilities if used as boot device		
Required		
15. Device Bay network adapter meets PC 98 requirements		
Required		
16. Internal ISDN device meets PC 98 network adapter requirements		
Required		
17. Internal ISDN device supports synchronous HDLC framing		
Required		
18. Internal ISDN device uses NDIS WAN miniport driver		
Required		
19. Internal ISDN device includes connection for analog phone		
Recommended		

Consumer PC 98	Office PC 98	Entertainment PC 98

20. ISDN device supports auto-SPID detection algorithms and standard SPID format
Required

21. ISDN driver supports switch detection
Recommended

22. ISDN driver supports unattended installation, with limitations
Required

23. ISDN device includes built-in NT-1
Recommended

24. ATM adapter meets PC 98 network adapter requirements
Required

25. ATM adapter supports a minimum number of active connections
Required

26. ATM adapter supports all service types defined by the ATM Forum
Recommended

27. ATM adapter supports a minimum number of simultaneously active rt-VBR/nrt-VBR/CBR connections
Required

28. ATM adapter supports traffic shaping
Required

29. ATM adapter supports external clocking
Required

30. ATM adapter supports OAM
Recommended

31. ATM adapter supports buffer chaining (Tx + Rx)
Recommended

32. ADSL device meets PC 98 network adapter requirements
Required

33. ATM/ADSL solution is implemented
Recommended

34. ADSL device supports RADSL
Recommended

35. Each device has a unique Plug and Play device ID
Required

36. Automatic resource assignment and dynamic disable capabilities are supported
Required

37. Plug and Play capabilities support multiple adapters
Required

38. All resource settings are reported in the user interface
Required

39. Device complies with device class power management reference specification
Required

Consumer PC 98	Office PC 98	Entertainment PC 98

40. Device supports wake-up events
Required

41. Device drivers and installation meet PC 98 requirements
Required

42. Driver supports promiscuous mode
Required

43. Driver works correctly with Microsoft network clients and protocols
Required

44. NDIS miniport driver does not make operating system–specific kernel calls
Required

45. NDIS 5.0 driver uses new INF format
Required

Checklist for Printers

1. IEEE 1394 printer meets PC 98 requirements for IEEE 1394
Required

2. USB printer meets PC 98 requirements for USB devices
Recommended

3. IEEE 1284 printer supports compatibility mode, nibble mode, and ECP, compliant with IEEE 1284-I
Required

4. IEEE 1284 printer meets IEEE 1284-II requirements
Recommended

5. ECP printer works correctly when ECP mode is turned off
Required

6. IEEE 1284 hardware supports error notification
Required

7. Implement Plug and Play support for all supported buses
Required

8. Peripheral device meets IEEE 1284 requirements
Required

9. Printer INF file and installation meet PC 98 requirements
Required

10. Driver correctly reports device capabilities
Required

11. Driver supports error notification
Required

12. Driver supports ICC color matching
Required

13. Port monitor software meets DDK guidelines
Required

14. Driver supports point-and-print network installation
Required

15. Device available immediately following installation
Required

16. Device supports accurate printable regions
Required

17. Driver supports required DDIs
Required

18. Driver based on unidriver
Recommended

Checklist for Scanners and Digital Cameras

1. Device uses PC 98 compatible port connection
Required

2. Icons provided for port and peripheral connectors
Required

3. Device supports ICC color matching
Required

4. IR device meets PC 98 IR requirements
Required

5. SCSI device meets PC 98 SCSI requirements
Required

6. SCSI device attaches to any PC 98-compliant SCSI controller
Required

7. USB device meets PC 98 USB requirements
Required

8. USB device supports string descriptors
Required

9. IEEE 1394 device meets PC 98 requirements for IEEE 1394
Required

10. Plug and Play capabilities implemented for all supported buses
Required

11. Each device has a Plug and Play device ID
Required

12. Device supports power management requirements for its bus
Required

13. Device drivers and installation meet PC 98 requirements
Required

14. Driver support is implemented under Still Image architecture
Required

15. Applications provided with the device meet Win32 specifications
Required

Hardware Glossary

Acronyms and Abbreviations

ABR available bit rate

AC alternating current

ACPI Advanced Configuration and Power Interface

A/D analog to digital

ADA Americans with Disabilities Act

ADC analog-to-digital converter

ADSL Asymmetric Digital Subscriber Line

AEC acoustic echo cancellation

AGP Accelerated Graphics Port

ANSI American National Standards Institute

API application programming interface

APIC Advanced Programmable Interrupt Controller

APM Advanced Power Management

APS analog protection system

ARC advanced RISC computing

ASCII American Standard Code for Information Interchange

ATA AT Attachment

ATAPI ATA Packet Interface

ATM Asynchronous Transfer Mode

ATSC Advanced Television Systems Committee

AT#UD Unimodem diagnostics command

AUI Attachment Unit Interface

A/V audio/video

AVGA Advanced VGA

BAR base address register

BDA BIOS Data Area

BIOS basic I/O system

BNC Bayonet Nut Connector. (*Also* British Naval Connector *or* Bayonet Neil Consulman)

BPB BIOS Parameter Blocks

bpp bits per pixel

bps bits per second

CAD computer-aided design

CBR constant bit rate

CDMA code division multiplexed access

CDPD cellular digital packet data

CHAP Challenge Handshake Authentication Protocol

CID CompatibleID

CIS card information structure

CMOS complementary metal-oxide semiconductor

COM (1) Component Object Model; (2) legacy serial port

CPTWG Copyright Protection Technical Working Group

CPU central processing unit

CRC cyclic redundancy check

CSEL Cable Select

CSN Card Select Number

CSR control and status register

CSS copy scramble system

CT Computer Telephony

D/A digital to analog

DAC digital-to-analog converter

dB decibel

DBC Device Bay Controller

DC direct current

DCE Data Communications Equipment

DDC display data channel

DDC2B *DDC Standard, Version 2.0*, Level B

DDI device driver interface

DDK Device Driver Kit

DES data encryption standard

DHCP Dynamic Host Configuration Protocol

DIB device-independent bitmap

DIP dual in-line package

DLL dynamic link library

DLS Downloadable Samples

DMA direct memory access

DMI Desktop Management Interface

DMTF Desktop Management Task Force

DRAM Direct Random Access Memory

DSP digital signal processor

DSS directory synchronization server

DSVD digital simultaneous voice/data

DTMF dual tone multifrequency

DTV digital television

DVB Digital Video Broadcast

DVC Digital Video Compression

DVD Optical disk storage that encompasses audio, video, and computer data

ECC error correction code

ECP extended capabilities port

ECR Engineering Change Request

EDID Extended Display Identification Data

EDT European Deaf Telephone

EGA enhanced graphics adapter

EIA Electronics Industries Association

8-VSB ATSC 8-Vestigial Side Band

EIO Extended I/O

EISA Extended Industry Standard Architecture

EMF enhanced metafile

EMI electromagnetic interference

EPG electronic program guide

EPP enhanced parallel port

ESCD Extended System Configuration Data

ETSI European Telecommunications Standards Institute

ExCA Exchangeable Card Architecture

FAT file allocation table

FAQ frequently asked questions

FCC Federal Communications Commission

FDC floppy disk controller

FDDI Fiber Distributed Data Interface

FEC forward error correction

FIFO first in/first out

FM frequency modulation

FP floating point

fps frames per second

FS A *See Glossary.*

FSIP full-scale input

FSK Frequency Shift Keyed

FSOP full-scale output

FTP file transfer protocol

GART Graphics Address Re-mapping Table

GDI Graphics Device Interface

GIDEI General Input Device Emulation Interface

GSM global system for mobile communications

HCL Hardware Compatibility List

HCT Hardware Compatibility Tests

HDLC high-level data link control

HDTV high-definition television

HEL hardware emulation layer

HFC hybrid fiber-coax

HID Human Interface Device

HRTF Head Related Transfer Function

HSCDS high-speed cable data services

Hz Hertz

IAL Intel Architecture Labs

ICC International Color Consortium

ICM Image Color Matching

ID identifier

IDE Integrated Device Electronics

IEC International Electrotechnical Commission

IEEE Institute for Electrical and Electronics Engineers

IETF Internet Engineering Task Force

IF Implementers Forum

IHV independent hardware vendor

I/O input/output

IOCTL I/O control

IP Internet Protocol

IPL Initial Program Load

IPX Internetwork Packet Exchange

IR infrared

IrDA Infrared Data Association

IRP I/O request packet

IRQ interrupt request

ISA Industry Standard Architecture

ISDN Integrated Service Digital Network

ISO International Standards Organization

ISO/OSI International Standards Organization Open Systems Interconnection

ISP Internet service provider

ISV independent software vendor

I2O intelligent I/O

ITU International Telecommunication Union

IVR interactive voice response

K kilobyte

Kbps kilobytes per second

Kss kilo-samples per second

L2 Level 2

LAN local area network

LBA logical block addressing

LCD liquid crystal display

LED light-emitting diode

LPCM location PCM

LPT line printer

LSB least significant bit

LUN logical unit number

LVE Live Video Extensions

m meter

MAC Media Access Control

MB megabyte

Mb/s megabits per second

MDK Modem Developers Kit

MEI Matsushita Electronics Incorporated

MESN Media Status Event Notification

MIDI Musical Instrument Digital Interface

MIP Multimission Interactive Picture

MP@ML Main Profile at Main Level

MPEG Moving Picture Expert Group

ms millisecond

MSB most significant bit

MSCDEX Microsoft CD-ROM Extensions

MSDN Microsoft Developer Network

MUX multiplex

NABTS North American Basic Teletext

NDIS Network Driver Interface Specification

NetBEUI NetBIOS Extended User Interface

Net PC Network PC

NICAM Near-Instantaneously Companded Audio Multiplex

NIDRR National Institute for Disability and Rehabilitation Research

NIUF National ISDN User's Forum

NMI Nonmaskable Interrupt

nrt non-real time

NTFS Windows NT file system

NT-1 network terminator

NTSC National Television System Committee

OAM operation and maintenance

OEM original equipment manufacturer

OFDM Orthogonal Frequency Division Multiplexing

OpenHCI Open Host Controller Interface

OR *See Glossary.*

OSR OEM service release

PAL Phase Alternation Line

PC personal computer

PCI Peripheral Component Interconnect

PCIC PC Card I/O cards

PCI PM *PCI Bus Power Management Specification, Revision 1.0* or higher

PCI 2.1 *PCI Local Bus Specification, Revision 2.1*

PCM pulse coded modulation

PCMCIA Personal Computer Memory Card International Association

PCR peak cell rate

PDA Personal Digital Assistant

PIC programmable interrupt controller

PID program ID

PIO programmed I/O

PIT programmable interrupt timer

PME power management event

POST power-on self-test

POTS plain old (analog) telephone service/system

PPP point-to-point protocol

PRI primary rate interface

PS/2 Personal System/2

PSTN Public Switched Telephone Network

PTT Post, Telephone, and Telegraph

QOS quality of service

RADSL rate adaptive digital subscriber line

RAID redundant array of inexpensive disks

RAM random-access memory

RAMDAC RAM digital-to-analog converter

RF radio frequency

RFC request for comments

RFP request for proposals

RISC reduced instruction set computing

RM resource management

RNA remote network access

ROM read-only memory

RP recommended practice

RSA public-key cipher for encryption/decryption

RSX Intel Realistic Sound eXperience

rt real time

SAP (1) Service Access Protocol;
(2) secondary audio programming

SAR segmentation and re-assembly

SCAM SCSI Configured Automatically

SCART Solent Club for Amateur Radio and Television; *also called* Peritel

SCID service channel ID

SCL system clock line

SCSI small computer system interface

SDK Software Developers Kit

SFF Small Form Factor

SIG Special Interest Group

SIR Serial IR

SIT Special Information Tone

SMART Self-Monitoring, Analysis, and Reporting Technology

SMBus System Management Bus

SMPTE Society of Motion Picture and Television Engineers

SNMP Simple Network Management Protocol

SNR signal-to-noise ratio

SOHO small office/home office

SPI Service Provider Interface

SPID service profile ID

SPX Sequenced Packet Exchange

SRC sample rate converter

STI Still Image architecture

STS/EN Status and Enable

SVGA Super VGA

TAM telephone answering machine

TAPI Telephony Application Program Interface

TCO total cost of ownership

TCP/IP Transmission Control Protocol/ Internet Protocol

TDD Telephone Device for the Deaf

TDMA time division multiplexed access

telset local telephone instrument

TFTP Trivial File Transfer Protocol

THD+N total harmonic distortion

3-D three-dimensional

TIA Telecommunications Industry Association

TP twisted pair

2-D two dimensional

UART Universal Asynchronous Receiver/Transmitter

UBR unspecified bit rate

UDF Universal Disk Format

UHCI Universal Host Controller Interface

UHF ultra-high frequency

unidriver universal printer driver

Unimodem universal modem driver

USB Universal Serial Bus

V volts

VBE VESA BIOS Extension

VBI vertical blanking interval

VBR variable bit rate

VCI virtual channel ID

VCR video cassette recorder

VDD Virtual display driver

VDIF Video Display Information Format

VDS virtual DMA services

VESA Video Electronics Standards Association

VfW Video for Windows

VGA video graphics array

VHF very high frequency

VLB VESA local bus

VPE Video Port Extensions

VPI virtual path ID

VpicD virtual PIC device

Vpp voltage point-to-point

VRML virtual reality modeling language

VRMS volts root-mean-square

VSD vendor specific driver

Vsync vertical synchronization

VxD virtual device driver

WAN wide area network

WBEM Web-Based Enterprise Management

WC write combining

WDL Windows Driver Library

WDM Win32 Driver Model

WfM Wired for Management

WHIIG *Windows Hardware Instrumentation Implementation Guidelines, Version 1.0*

WHQL Windows Hardware Quality Labs

WinSock Windows Sockets 2.0

WMI Windows Management Instrumentation

WSS Windows Sound System

YUV *See Glossary.*

ZV Zoomed Video

Glossary

A

AC-3 An audio standard developed by Dolby Laboratories for delivering 5.1 audio. This system compresses six channels of digital audio into 384 Kbps versus 4 Mb/s uncompressed.

ACPI Advanced Configuration and Power Interface. A specification that defines a new interface to the system board. This interface enables the operating system to implement operating system–directed power management and system configuration. ACPI allows system manufacturers to build systems consistent with the OnNow design initiative for instantly available PCs.

ACPI hardware Computer hardware with the features necessary to support operating system power management and with the interfaces to those features described using the Description Tables as specified in the ACPI specification.

adapter *See* device.

add-on devices Devices that are traditionally added to the base PC system to increase functionality, such as audio, networking, graphics, SCSI controller, and so on. Add-on devices fall into two categories: devices built onto the system board, and devices on expansion cards added to the system through a system-board connector such as PCI.

ADSL Asymmetric Digital Subscriber Line. A method for moving data over regular phone lines. An ADSL circuit is much faster than a regular phone connection, even though the wires coming into the subscriber's premises are the same (copper) as used for regular phone service.

analog A method of signal representation by an infinitely smooth universe of numeric values. Measurements that are characterized as analog include readings of voltage and current. *Compare with* digital.

analog video A video signal that represents an infinite number of smooth gradations between given video levels. *Compare with* digital video.

API Application programming interface. A set of routines that an applications program uses to request and carry out lower-level services performed by a computer operating system.

APM Advanced Power Management. A software interface (defined by Microsoft and Intel) between hardware-specific power management software (such as that located in a system BIOS) and an operating system power management driver.

architecture A general term referring to the structure of all or part of a computer system. Also refers to the design of system software, such as the operating system, as well as to the combination of hardware and basic software that links machines on a computer network.

ASCII American Standard Code for Information Interchange. The most popular coding method used by small computers for converting letters, numbers, punctuation, and control codes into digital format.

ATA AT Attachment. A compatible register set, and a 40-pin connector and its associated signals. More commonly known as IDE.

ATAPI ATA Packet Interface. A hardware and software specification that documents the interface between a host computer and the CD-ROM drives using the ATA bus.

ATM Asynchronous transfer mode. A transmission protocol that segments user traffic into small, fixed-size units called cells that are transmitted to their destination, where they are reassembled into the original traffic. During transmission, cells from different users may be intermixed asynchronously to maximize utilization of network resources.

AUI Attachment Unit Interface. The portion of the Ethernet standard that specifies how a cable is to be connected to an Ethernet card. AUI specifies a cable connected to a transceiver that plugs into a 15-pin socket on the network adapter.

B

BIOS Basic I/O system. A set of routines that works closely with the hardware to support the transfer of information between elements of the system, such as memory, disks, and the monitor. Although critical to performance, the BIOS is usually invisible to the end user; however, programmers can access it.

BNC Bayonet Nut Connector. *Also* British Naval Connector *or* Bayonet Neil Consulman. A type of connector used with coaxial cables such as the RG-58 A/U cable used with the 10Base-2 Ethernet system.

C

CD-ROM Compact disc read-only memory. A 4.75-inch laser-encoded optical memory storage medium (developed by NV Philips and Sony Corporation) with the same constant linear velocity (CLV) spiral format as compact audio discs and some video discs. CD-ROMs can hold about 550 MB of data.

class For hardware, the manner in which devices and buses are grouped for purposes of installing and managing device drivers and allocating resources. The hardware tree is organized by device class.

class driver A driver that provides system-required, hardware-independent support for a given class of physical devices. Such a driver communicates with a corresponding hardware-dependent port driver, using a set of system-defined device control requests, possibly with additional driver-defined device control requests. Under WDM, the class driver creates a device object to represent each adapter registered by minidrivers. The class driver is responsible for multiprocessor and interrupt synchronization.

codec Coder-decoder. A filter that manipulates data in some form, usually by compressing or decompressing the data stream.

COM (1) Component Object Model; the core of OLE (object linking and embedding). Defines how OLE objects and their clients interact within processes or across process boundaries. (2) Legacy serial port.

concatenate To join sequentially.

configuration manager The Windows Plug and Play system component that drives the process of locating devices, setting up their nodes in the hardware tree, and running the resource allocation process. Each of the three phases of configuration management—boot time, real mode, and protected mode—have their own configuration managers.

controllerless modem *Also* host-based controller. A modem that consists of a DSP without the usual microcontroller. The host CPU provides the AT command interpreter, modem-control functions, and v.42bis implementation. *Compare with* software modem.

control method A definition of how an ACPI-compatible operating system can perform a simple hardware task. For example, the operating system invokes control methods to read the temperature of a thermal zone. Control methods are written in an encoded language called AML (ACPI Machine Language).

CPU Central processing unit. A computational and control unit of a computer; the device that interprets and executes instructions. By definition, the CPU functions as the “brain” of the computer.

CSN Card Select Number. The handle created by the system BIOS or the operating system through the isolation process and assigned as a unique ID to each Plug and Play card on the ISA bus.

D

DDC Display data channel. The Plug and Play baseline for monitors. The communications channel between a monitor and the display adapter to which it is connected. This channel provides a method for the monitor to convey its identity to the display adapter.

device Any circuit that performs a specific function, such as a parallel port.

Device Bay An industry specification that defines a mechanism for both peripheral devices and system bays. Allows adding and upgrading PC peripheral devices without opening the chassis.

device ID A unique ASCII string for a device created by enumerators to identify a hardware device and used to cross-reference data about the device stored in the registry. Distinguishes each logical device and bus from all others on the system.

device node *Also* devnode. The basic data structure for a given device, built by the configuration manager. Device nodes are built into memory at system start-up for each device and enumerator. Each device node contains information about the device, such as currently assigned resources.

device object A kernel-mode–only object type used to represent a physical, logical, or virtual device whose driver has been loaded into the system.

devnode *See* device node.

digital A method of signal representation by a set of discrete numerical values, as opposed to a continuously fluctuating current or voltage. *Compare with* analog.

digital video A video signal represented by computer-readable binary numbers that describe a finite set of colors and luminance levels. *Compare with* analog video.

disk I/O controller *Also* hard disk controller. A special-purpose chip and circuitry that directs and controls reading from and writing to a computer’s disk drive.

DLL Dynamic link library. API routines that user-mode applications access through ordinary procedure calls. The code for the API routine is not included in the user’s executable image. Instead, the operating system automatically points the executable image to the DLL procedures at run time.

DMA Direct memory access. A method of transferring data between peripheral and host memory without processor intervention. The system board uses a DMA controller to handle a fixed number of channels, each of which can be used by only one device at a time.

DMI Desktop Management Interface. A framework created by the DMTF. DMTF specifications define industry-standard interfaces for instrumentation providers and management applications.

docking station The base computer unit into which a user can insert a portable computer, expanding it to a desktop equivalent. A typical docking station provides drive bays, expansion slots, all the ports on an equivalent desktop computer, and AC power.

driver Kernel-mode code used either to control or emulate a hardware device.

DSP Digital signal processor. An integrated circuit designed for high-speed data manipulations. Used in audio, communications, image manipulation, and other data-acquisition and data-control applications.

DTV Digital television. DTV standards allow standard resolution mode—with about twice the horizontal resolution of conventional analog broadcasts—as well as HDTV mode. Video uses MPEG-2 digital compression, and audio uses AC-3 (Digital Dolby) compression.

DVD Optical disk storage that encompasses audio, video, and computer data.

E

ECP Extended capabilities port. An asynchronous, 8-bit–wide parallel channel defined by IEEE 1284-1944 that provides PC-to-peripheral and peripheral-to-PC data transfers.

EISA Extended Industry Standard Architecture. A 32-bit PC expansion bus designed as a superset of the ISA bus. Designed to expand the speed and data width of the legacy expansion bus while still supporting older ISA cards.

embedded controller The general class of microcontrollers used to support OEM-specific implementations, mainly in mobile environments. The embedded controller performs complex low-level functions through a simple interface to the host microprocessor.

embedded controller interface ACPI defines a standard hardware and software communications interface between an operating system driver and an embedded controller—for example, Smart Battery and AML code. This allows any operating system to provide a standard driver that can directly communicate with an embedded controller in the system, thus allowing other drivers to communicate with and use the resources of system embedded controllers.

EPG Electronic program guide. The on-screen user interface that allows users to select, manage, and search television programs and other content-viewing options.

expansion bus A group of control lines that provide a buffered interface to devices located either on the system board or on cards that are plugged into expansion connectors. Common expansion buses included on the system board are USB, PC Card, and PCI.

expansion card A card that connects to an expansion bus and contains one or more devices.

expansion ROM *See* option ROM.

F

FDC Floppy disk controller. A special-purpose chip and associated circuitry that directs and controls reading from and writing to a computer's disk drive.

FIFO First in/first out. A method for processing a queue in which items are removed in the same order in which they were added.

FS A Decibels relative to full scale, measured using "A weighting" filters.

G

GUID Globally unique ID. A 16-byte value generated from the unique ID on a adapter, the current date and time, and a sequence number. This is used to allow any party to create IDs that will be guaranteed not to overlap with other similarly created IDs.

H

hardware branch The hardware archive root key in the registry that is a superset of the memory-resident hardware tree. Although the hardware tree contains information only about those devices currently detected and running in the system, the registry contains a complete list of all hardware ever installed on the particular computer. The hardware root key is \\Hkey_Local_Machine\Hardware.

hardware tree A record in RAM of the current system configuration, based on the information for all devices in the hardware branch of the registry. The hardware tree is created each time the system is started or whenever a dynamic change occurs in the system configuration.

HCI Host controller interface. For example, a system-level interface supporting USB.

HCL Hardware Compatibility List. A registry of products that have been tested by WHQL and that have passed Windows compatibility testing.

HCT Hardware Compatibility Tests. A suite of tests from WHQL that verifies hardware and device driver operations under a specific operating environment. These tests exercise the combination of a device, a software driver, and an operating system under controlled conditions to verify that all components operate properly.

HDTV High-definition television. A proposed standard that recommends doubling the current 525 lines per picture to 1050 lines, and increasing the screen aspect ratio (that is, width to height) from the current 12:9 to 16:9, which would create a television screen shaped more like a movie screen.

HID specification The device class definition developed by the USB standards group for Human Interface Devices. Serves as the basis for WDM input device support, and unifies input devices by providing flexible data reporting, typeless data, and arrayed and variable input and output.

I

ID Identifier. Generally, any text string used as a label, such as the name of a procedure or a variable in a program, or the name attached to a hard drive or floppy disk.

IDE Integrated Device Electronics. A type of disk drive interface where the controller electronics reside on the drive itself, eliminating the need for a separate adapter card.

IEEE Institute of Electrical and Electronics Engineers, pronounced "I-triple-E." Founded in 1963, IEEE is an organization composed of engineers, scientists, and students. IEEE is best known for developing standards for the computer and electronics industry.

INF file Information file. A file created for a particular adapter that provides the operating system with information required to set up a device, such as a list of valid logical configurations for the device, the names of driver files associated with the device, and so on. The device manufacturer typically provides an INF file on a disk with an adapter.

INI file Initialization file. Commonly used under Windows 3.*x* and earlier, INI files have been used by both the operating system and individual applications to store persistent settings related to an application, driver, or piece of hardware. In Windows and Windows NT, INI files are supported for backward compatibility, but the registry is the preferred location for storing such settings.

I/O Input/output. Two of the three activities that characterize a computer (input, processing, and output). Refers to the complementary tasks of gathering data for the microprocessor to work with and making the results available to the user through a device such as the display, disk drive, or printer.

IOCTL I/O control. A custom class of IRPs available to user mode. Each WDM class driver has a set of IOCTLs that it uses to communicate with applications. The IOCTLs give the class driver information about intended usage by applications. The class driver performs all IOCTL parameter validation.

IPL Initial program load. A device used by the system during the boot process to load the operating system into memory.

IRP I/O request packet. Data structures that drivers use to communicate with each other. The basic method of communication between kernel-mode devices. An IRP is a key data structure for WDM, which features multiple layered drivers.

IRQ Interrupt request. A method by which a device can request to be serviced by the device's software driver. The system board uses a PIC to monitor the priority of the requests from all devices. When a request occurs, a microprocessor suspends the current operation and gives control to the device driver associated with the interrupt.

ISA Industry Standard Architecture. An 8-bit (and later, a 16-bit) legacy expansion bus that provides a buffered interface from devices on expansion cards to the PC internal bus.

ISDN Integrated Service Digital Network. A set of communications standards that enables a single phone line or optical cable to carry voice, digital network services, and video.

ISR Interrupt service routine. A routine whose function is to service a device when it generates an interrupt.

L

LAN Local area network. A group of computers and other devices dispersed over a relatively limited area and connected by a communications link that enables any device to interact with any other device on the network. *Compare with* WAN.

LBA Logical block address. A unit of data supplied or requested by a host computer.

legacy Any feature in the system based on older technology for which compatibility continues to be maintained in other system components.

M

Microsoft DirectShow *Formerly* ActiveMovie. A cross-platform API for developers of multimedia applications that provides a user-mode connection and Stream architecture to support high-quality digital video, high-fidelity audio, and special effects.

Microsoft DirectX A low-level API that provides user-mode media interfaces for games and other high-performance multimedia applications. DirectX is a thin layer, providing direct access to hardware services. DirectX takes advantage of available hardware accelerators and emulates accelerator services when accelerators are not present.

MIDI Musical Instrument Digital Interface. An industry-standard connection for computer control of musical instruments and devices. A hardware and data standard for communicating between hardware. Most references involve only the data standard, which is a byte stream used for controlling musical instruments and storing the output of such instruments.

minidriver A hardware-specific DLL that uses a Microsoft-provided class driver to accomplish most actions through functions call and provides only device-specific controls. Under WDM, the minidriver uses the class driver's device object to make system calls.

miniport driver A device-specific kernel-mode driver linked to a Windows NT or WDM port driver, usually implemented as a DLL that provides an interface between the port driver and the system.

motherboard *See* system board.

MPEG Moving Picture Expert Group. Refers to one of several standard video-compression schemes. A codec for squeezing full-screen, VHS-quality digital video into a small data stream so that it can be played from a CD-ROM drive.

multifunction device A piece of hardware that supports multiple, discrete functions, such as audio, mixer, and music, on a single adapter.

multimedia Refers to the delivery of information that combines different content formats, such as motion video, audio, still image, graphics, animation, text, and so forth.

N

NDIS Network Driver Interface Specification. The interface for network drivers used in Windows and Windows NT operating systems. NDIS provides a common mechanism by which any given NDIS-compatible transport driver can communicate with any NDIS-compatible network adapter driver. Moreover, it provides for multiple transports to work over multiple network adapters by supporting multiplexing between transports and drivers.

Net PC Network PC. A PC designed to meet the industry specification for Network PC systems, which optimizes PC design for flexibility and manageability in order to reduce the total cost of ownership (TCO).

NMI Nonmaskable Interrupt. An interrupt that cannot be overruled by another service request. A hardware interrupt is called nonmaskable if it cannot be masked by the processor interrupt flag.

NTSC National Television System Committee of the Electronics Industries Association (EIA). The standards-setting body for television and video in the United States. Sponsor of the NTSC standard for encoding color, a coding system compatible with black-and-white signals and the first system used for color broadcasting in the United States. The broadcast standard for the United States and Japan. *See also* PAL format *and* SECAM.

NTSC format A color-television format having 525 scan lines, a field frequency of 60 Hz, a broadcast bandwidth of 4 MHz, line frequency of 15.75 KHz, frame frequency of 1/30 of a second, and a color subcarrier frequency of 3.58 MHz. *See also* PAL format *and* SECAM.

O

OEM Original equipment manufacturer. Used primarily to refer to PC systems manufacturers.

OnNow A design initiative that seeks to create all the components required for a comprehensive, system-wide approach to system and device power control. OnNow is a term for a PC that is always on but appears off and that responds immediately to user or other requests.

OpenGL An operating system independent, industry-standard API for 3-D color graphics programming. Typically used for engineering, visualization, simulation, and other graphics-intensive applications.

option ROM *Also* expansion ROM. Optional read-only memory found on an expansion card. Option ROMs usually contain additional firmware required to properly boot the peripheral connected to the expansion card, for example, a hard drive.

OR A logical operation for combining two bits or two Boolean values. If one or both values are true, it returns the values of true. *Compare with* XOR.

P

PAL format Phase Alternation Line format. The European video standard, except for France. *See also* NTSC *and* SECAM.

PC 97 The 1997–98 requirements for PC system and peripheral design for the "Designed for Microsoft Windows" logo, as defined in *PC 97 Hardware Design Guide.*

PC 98 The 1998–99 requirements for PC system and peripheral design for the "Designed for Microsoft Windows" logo, as defined in *PC 98 System Design Guide,* which is an addendum to *PC 97 Hardware Design Guide.*

PC Card A trademark of PCMCIA. A removable device that is designed to be plugged into a PCMCIA slot and used as a memory-related peripheral.

PCI Peripheral Component Interconnect. A high-performance, 32-bit or 64-bit bus designed to be used with devices that have high bandwidth requirements, such as a display subsystem.

PCM Pulse coded modulation. A method of encoding information in a signal by varying the amplitude of pulses. The most common method of encoding an analog signal into a digital bit stream, usually 16 bits per sample.

PCMCIA Personal Computer Memory Card International Association. Sometimes used to refer to a controller for a type of expansion card documented in the PCMCIA standards.

Plug and Play A design philosophy and set of specifications that describe hardware and software changes to the PC and its peripherals that automatically identify and arbitrate resource requirements among all devices and buses on the system. Plug and Play specifies a set of API elements that are used in addition to, but not in place of, existing driver architectures.

Plug and Play BIOS A BIOS with responsibility for configuring Plug and Play cards and system-board devices during system power up. Provides run-time configuration services for system-board devices after start-up. *See also* ACPI.

power management Mechanisms in software and hardware to minimize system power consumption, to manage system thermal limits, and to maximize system battery life. Power management involves trade-offs among system speed, noise, battery life, processing speed, and power consumption.

push technology In client/server applications, to send data to a client without the client requesting it—for example, sending e-mail. In contrast, the World Wide Web is based on a pull technology, where the client browser must request a web page before it is sent. Broadcast media are push technologies because they send information out regardless of whether anyone is tuned in.

R

RAM Random access memory. Semiconductor-based memory that can be read and written by the microprocessor or other hardware devices.

RAMDAC RAM digital-to-analog converter. A chip built into some VGA and SVGA display adapters that translates the digital representation of a pixel into the analog information needed by the monitor to display it.

rasterization The conversion of vector graphics (images described mathematically as points connected by straight lines) to equivalent images composed of pixel patterns that can be stored and manipulated as sets of bits.

Red Book audio The data format standard for conventional audio CDs used in home stereo systems.

registry In Windows and Windows NT, the tree-structured hierarchical database where general system hardware and software settings are stored. The registry supersedes the use of separate INI files for all system components and applications that know how to store values in the registry.

resource (1) Any sort of set from which a subset can be allocated for use by a client, such as memory or bus bandwidth. This is not the same as resources that are allocated by Plug and Play. (2) A general term that refers to IRQ signals, DMA channels, I/O port addresses, and memory addresses for Plug and Play.

resource conflict In Plug and Play device configuration, the result of more than one device sharing a nonshareable resource. Conflicts can cause the device to be partially functional or nonfunctional, orz can cause the PC to malfunction completely.

RISC Reduced instruction set computing. A type of microprocessor design that focuses on rapid and efficient processing of a relatively small set of instructions. RISC architecture limits the number of instructions that are built into the microprocessor, but optimizes each so it can be carried out very rapidly—usually within a single clock cycle.

RISC-based Refers to computers based on Windows NT-compatible implementations of RISC processors.

S

SCI System control interrupt. A system interrupt used by hardware to notify the operating system of ACPI events. The SCI is an active low, shareable, level interrupt.

SCSI Small computer system interface, pronounced "scuzzy." An I/O bus designed as a method for connecting several classes of peripherals to a host system without requiring modifications to generic hardware and software.

sealed case A PC system design that does not provide end-user–accessible internal expansion slots. This is the equivalent of "no user-serviceable parts inside" for consumer appliances. A sealed case can provide external expansion capabilities.

SECAM Sequential Couleur a Memoire (Sequential Color with Memory). The television standard for France, Russia, and most of Eastern Europe. As with PAL, SECAM is based on a 50-Hz power system, but it uses a different encoding process and displays 819 horizontal lines per frame at a scan rate of 25 frames per second (50 fields per second). *See also* NTSC *and* PAL format.

SIPC Simply Interactive PC. A vision guiding investments that Microsoft is making in software and hardware advances to make the PC as simple, convenient, and approachable as an appliance.

SMBus System Management Bus. A two-wire interface based on the I²C protocol. The SMBus is a low-speed bus that provides positive addressing for devices, as well as bus arbitration.

software device A filter in kernel streaming and DirectShow (formerly ActiveMovie) that has no underlying hardware associated with it.

software modem *Also* host-based signal processing or pumpless modem. Performs signal processing on the host CPU, and implements the controller using V.42bis. The modem hardware consists of a telephone-line interface and digital-to-analog and analog-to-digital conversion circuitry. The hardware does not contain a DSP or a microcontroller. *Compare with* controllerless modem.

Sound Blaster Hardware produced by Creative Labs, Inc., that represents for MS-DOS–based games one of the major hardware interfaces for both audio and music (specifically MIDI) data.

SPI Service Provider Interface. Component in Microsoft networking, TAPI, and other communications technologies.

spin down A power-management capability in which a hard drive shuts down its spindle motor.

Still Image architecture *Also* STI. A WDM architecture for still-image devices. A still-image minidriver provides support for still-image devices such as scanners and cameras under the WDM Still Image architecture.

SVGA Super VGA. A video standard established by VESA to provide high-resolution color display on IBM-compatible computers. The most common SVGA standard is 1024 × 768 pixels resolution.

S-Video *Also* Y/C video. A video signal that separates the luminance and color (Y and C) components of the signal for improved quality over composite video. The type of video signal used in the Hi8 and S-VHS videotape formats. Transmits luminance and color portions separately, using multiple wires, thus avoiding the NTSC encoding process and its inevitable loss of picture quality.

system board *Also* motherboard *or* planar. The primary circuit board in a PC that contains most of the basic components of the system.

system devices Devices on the system board, such as interrupt controllers, keyboard controller, real-time clock, DMA page registers, DMA controllers, memory controllers, FDC, IDE ports, serial and parallel ports, PCI bridges, and so on. These devices are typically integrated into the supporting chip set.

T

TAPI Telephony API. A set of Win32-based calls that applications use to control modems and telephones by routing application function calls to the appropriate service-provider DLL for a modem.

telephony Telephone technology.

tuple A data structure defined by PCMCIA to describe a single, specific characteristic of a PC Card device. Tuples are chained together to form the CIS, which describes to system software the PC Card's resource requirements and other characteristics. Tuples consist of a tuple code, an offset to the next tuple, and a number of bytes specific to the tuple.

TWAIN API for image acquisition developed by an association of industry leaders. The *TWAIN Specification, Version 1.6* or higher, is available from http://www.twain.org.

U

UART Universal Asynchronous Receiver/Transmitter. A module composed of a circuit that contains both the receiving and transmitting circuits required for asynchronous serial communication.

Unimodem Universal modem driver. A driver-level component that uses modem description files to control its interaction with the communications driver.

UPS Uninterruptible power supply. A device connected between a computer and a power source that ensures that electrical flow to the computer is not interrupted because of a blackout and, in most cases, protects the computer against potentially damaging events such as power surges.

USB Universal Serial Bus. A bi-directional, isochronous, dynamically attachable serial interface for adding peripheral devices such as game controllers, serial and parallel ports, and input devices on a single bus.

user mode The nonprivileged processor mode in which application code executes, including protected subsystem code in Windows NT.

V

VAR Value added reseller or retailer. A company that resells hardware and software packages made by another company (such as an OEM) with extra components added (such as specialist software) to developers and/or end users.

VBI Vertical blanking interval. The time interval between television fields needed for the scanning gun to move from the bottom of the screen to the top for the start of the next field.

VCACHE In Windows, a 32-bit protected-mode cache driver.

VCOMM In Windows, a 32-bit protected-mode communications driver.

VCR Video cassette recorder. An analog magnetic recording and playback machine. Typically used for recording and viewing full-motion video. Also useful as a data backup device.

VGA Video graphics array. A video adapter that supports 640 × 480-pixel color resolution. A video display standard for boot devices under Windows operating systems.

VM Virtual machine. Software that mimics the performance of a hardware device.

VPE Video Port Extensions. Extensions to the DirectDraw API to control the video stream from the video port within the context of VGA memory.

VxD Virtual device driver. A device driver that runs at the privileged ring 0 protected mode of the microprocessor. Can extend the services of the Windows kernel, supervise hardware operations, or perform both functions. Such driver files are usually named according to the scheme VxD, where *x* refers to the device or service supported.

W

WAN Wide area network. A communications network that connects geographically separated areas. *Compare with* LAN.

warm docking A method of removing or installing a mobile system in a docking station by which the computer can be docked or undocked while in a reduced power state, such as suspend.

WBEM Web-based Enterprise Management. Technology based on standards being developed by DMTF and IETF. WBEM will provide a mechanism to specify information exchange between management applications and managed components.

WDL Windows Driver Library. *See* WHQL.

WDM Win32 Driver Model. A 32-bit driver model based on the Windows NT driver model that is designed to provide a common architecture of I/O services for both Windows and Windows NT for specific classes of drivers. These driver classes include USB and IEEE 1394 buses, audio, still-image capture, video capture, and HID-compliant devices such as USB mice, keyboards, and joysticks. Provides a model for writing kernel-mode drivers and minidrivers, and provides extensions for Plug and Play and power management.

WHQL Windows Hardware Quality Labs. Provides compatibility testing services to test hardware and drivers for Windows NT and Windows. Administers testing for the "Designed for icrosoft Windows" logo programs. Author of WDL and HCL. For more information, see the web site at http://www.microsoft.com/hwtest/.

Win32 API A 32-bit application programming interface for both Windows and Windows NT that includes operating system capabilities, security, and API routines for Windows-based applications.

Windows Refers to the Microsoft Windows 98 operating system, including any add-on capabilities and any later versions of the operating system.

Windows NT Refers to the Microsoft Windows NT version 5.0 operating system, including any add-on capabilities and any later versions of the operating system, unless specific design issues are defined that relate to version 5.0.

Windows NT DDK Documents the Windows NT driver model (upon which WDM is based) and is an essential component for building WDM drivers. Provided through MSDN Professional membership.

Windows NT driver model The layered device driver model used under the Windows NT operating system. For information, see *Inside Windows NT*, by Helen Custer (Microsoft Press, 1993; ISBN 1 -55615-481-X).

WMI Windows Management Instrumentation. Extensions to WDM developed for Windows NT 5.0 and Windows 98 to provide an operating system interface through which instrumented components can provide information and notifications.

workstation In general, a powerful computer with considerable calculating and graphics capabilities.

X

XOR Exclusive OR. A Boolean operation that yields "true" if and only if one of its operands is true and the other is false. *Compare with* OR.

Y

YcrCb *See* YUV.

YUV The method of color encoding for transmitting color video images while maintaining compatibility with black-and-white video. Uses less bandwidth than the three separate video signals in an RGB video transmission. Consists of two major components: luminance (Y), which corresponds to the brightness of an image pixel, and chrominance (UV or CrCb), which corresponds to the color of an image pixel.

Z

Zero Administration initiative for Windows *Also* Zero Administration initiative. An initiative that focuses on improving Windows and Windows NT for maximum automation of administrative tasks with centralized control and maximum flexibility.

Index

Entries in italics refer to titles of standards and specifications.

B

C

D

E

I

J

K

O

P

Q

R

T

U

V

W

X

Y

Z