

CHAPTER 1

PC 97 Design Issues

“PC 97” represents the collection of PC system definitions and bus and device design requirements that make up the next generation of guidelines for the “Designed for Microsoft Windows” logo.

This chapter introduces the PC 97 design issues for computer systems that run the Microsoft Windows family of operating systems for x86-based PCs and that run Windows NT on the RISC-based processor platform.

For more information about design issues related to the OnNow design initiative and Win32 Driver Model, see the “OnNow and WDM for PC 97” chapter.

Important The system requirements defined in *PC 97 Hardware Design Guide* provide guidelines for designing PC systems that best run Windows 95 and Windows NT. These design requirements are not related to the basic system requirements for running a Windows operating system. For information about the basic system requirements for running Windows 95 and Windows NT, see:

<http://www.microsoft.com/windows/mix/>

Contents

Evolving the PC Platform	2
Logo Program at Microsoft for Hardware	3
Designing PCs for Windows 95 and Windows NT	5
Designing with New Bus and Device Support	6
SIPC and Designing for Ease of Use	7
Plug and Play and Designing for Extensibility	9
Designing for x86-based and RISC-based PCs	11
References for PC 97 Design Issues	13

Evolving the PC Platform

Every member of the PC industry has an important business goal that involves growing the PC market among business and home users. An important key to growing any part of the PC hardware market is to increase the ease of use for PCs in general and to provide support for new kinds of hardware in particular:

- Business customers need improved ease-of-use capabilities for hardware, both to increase end-user productivity and to lower support costs, resulting in reduced total cost of ownership.
- Home users need easier-to-use hardware to overcome barriers to learning and using unfamiliar technology.
- All users need hardware and operating-system support for new technologies that will lower costs, improve usability, and improve performance for communications, applications, and media titles.

The endeavor to make PC hardware easier to use can only be accomplished as an industry partnership among hardware vendors, PC manufacturers, and Microsoft to design hardware, operating systems, and software that work together effectively. One of the most exemplary efforts for this endeavor in the recent past has been the Plug and Play initiative that the industry has embraced.

To continue evolving both the usability and new capabilities for PC hardware, the industry must continue to work on evolving the PC platform in key ways:

- To enhance the usability and supportability of existing and new hardware devices in ways that lower everyone's support costs.
- To broaden consumer PC capabilities so that the PC becomes a center for entertainment, communications, and home productivity.

Microsoft is dedicated to working in strategic industry relationships that deepen and strengthen support for evolving the PC platform. To this end, Microsoft is involved in these efforts:

- Working with industry groups to define standards for new technologies.
- Designing operating-system support for new bus and device classes to ensure new technologies can reach a broad market quickly.
- Enhancing the Windows operating systems to make it easy for both hardware and software developers to exploit operating system capabilities.
- Offering the "Designed for Microsoft Windows" logo and testing program to enable users to identify hardware and software designed or take advantage of the Windows operating systems.

The intent of the PC 97 Hardware Design Guidelines and the “Designed for Microsoft Windows” program is to make PCs, hardware components, and software easier to use. The system design requirements defined in this guide support a synergy among PC hardware, Microsoft Windows operating systems, and Windows-based software. The new PC 97 requirements for systems and components are based on the following high-level goals:

- Support for Plug and Play compatibility and power management for configuring and managing all system components, including new requirements under the OnNow design initiative.
- 32-bit device drivers and installation procedures for both Windows 95 and Windows NT written as defined in the Windows 95 DDK or the Windows NT DDK.
- Compliance with standards for system platform, buses, and devices:
 - Buses and devices meet industry specifications for each bus type.
 - Devices meet industry standards and specifications for each device class.
 - System and devices meet minimal performance standards.
 - System and devices meet guidelines for ease of use and industrial design.

Logo Program at Microsoft for Hardware

The “Designed for Microsoft Windows” logo on PC hardware provides end users with the assurance that this hardware will work well with Windows and that it takes advantage of features built in to the operating system. Products that receive this logo go through a rigorous series of tests administered by the Windows Hardware Quality Labs (WHQL) to determine that the hardware meets all “Designed for Microsoft Windows” criteria.

The first “Designed for Microsoft Windows 95” logo program was based on requirements defined specifically for Windows 95. Under the requirements for the first logo program, part of the logo testing requirements have been that a product be tested under Windows NT, with the requirement being that the product must run successfully on both Windows 95 and Windows NT unless prevented by architectural differences between the two operating systems. This guide defines the specific differences for drivers or other support under the two operating systems.

The collection of categories for PC 97 system types are the following:

- **Basic PC 97:** Defines a standard PC (both desktop and mobile) that sets a baseline of features and performance for users in all categories. The Basic PC 97 system is designed to best support typical Windows-based applications running under either Windows 95 or Windows NT Workstation. However, these requirements represent much more than the minimum system requirements for the operating systems, which are much lower in order to support a large base of older, less capable hardware.
- **Workstation PC 97:** Defines enhanced components for a PC system designed to support the more complex requirements of Win32®-based applications that have strenuous graphics or computational demands. A Workstation PC 97 system can run either Windows 95 or Windows NT Workstation, but its design is optimized to support Windows NT Workstation when used for complex applications such as CAD or engineering computing, software development, and content authoring for the Internet or applications.
- **Entertainment PC 97:** Defines enhanced components for a PC system designed as an interactive multimedia system optimized for games, education, personal communications, Internet activities, and video playback. Key features that differentiate Entertainment PC 97 from other system types include support for advanced game playing 3-D graphics and audio, high-performance graphics and video including full motion MPEG-2 video, high-fidelity audio, and connections to consumer-electronics devices such as home stereos, telephones, VCRs, camcorders, digital satellite systems, and so on.

The PC 97 requirements for each system type are defined in Part 2 of this guide. In addition, there are additional specific logo criteria for each bus and device class, as defined in Part 3, “Bus Design for PC 97,” and Part 4, “Device Design Guidelines,” in this guide.

Notice that mobile PCs are not defined as a separate system category, as mobile PC designs could be implemented under each PC 97 system type. The “Basic PC 97” chapter presents a summary list of specific design issues for mobile PCs and docking stations. Also, PC 97 Hardware Design Guidelines do not apply for so-called personal digital assistants (PDAs) or other sub-PC form factor designs.

Microsoft will continue to have a single logo for all categories of computers defined for PC 97. However, for systems submitted under the Workstation and Entertainment PC categories, in 1997 Microsoft will be announcing and offering additional marketing programs for those specific categories.

Designing PCs for Windows 95 and Windows NT

The new guidelines presented in *PC 97 Hardware Design Guide* define PC system types and component requirements for the Microsoft Windows family of operating systems. This means that, for the first time, the “Designed for Microsoft Windows” program requirements are defined in one place for two operating systems:

- **Microsoft Windows 95.** Windows 95 is an operating system for platforms that use Intel and equivalent microprocessors (referred to as “x86-based systems” in this guide). Windows 95 is the successor to MS-DOS®, Windows version 3.1, and Windows for Workgroups version 3.x. Windows 95 includes built-in networking capabilities and broad support for legacy hardware and software, plus support for new hardware and new APIs.
- **Microsoft Windows NT Workstation and Windows NT Server.** Windows NT is a preemptive, multitasking operating system that includes security and networking services as fundamental components of the base operating system. Windows NT runs on both complex instruction set computing (CISC) and reduced instruction set computing (RISC) processors. Windows NT also supports high-performance computing by providing kernel support for computers that have symmetric multiprocessor configurations.

In 1996, Windows 95 and Windows NT 4.0 share such technologies as the Win32 API, OLE, networking, and user interface, so it is possible to run applications based on a common API set and a common object technology. Both products contain the essential infrastructure for desktop management by using a system registry that is accessible remotely using Win32 APIs.

In the future, these operating systems will share even more capabilities, including the Win32 Driver Model (WDM), the new Advanced Configuration and Power Interface (ACPI), and Plug and Play capabilities, as described in the “OnNow and WDM for PC 97” chapter. Other shared capabilities will include Microsoft DirectX™ APIs, ActiveX™ APIs, and distributed Component Object Model protocol (formerly referred to as “network OLE”). Because of differences in product development cycles, new technologies might appear on one product before the other, but these differences will be temporary.

All support for new technologies such as USB input devices and IEEE 1394 digital video devices will use WDM. To take advantage of new support in both operating systems, the hardware manufacturer can use the built-in drivers or create only a minidriver to exploit specific features in the hardware.

Although more code will be shared on both platforms over time, there will continue to be two complementary desktop operating system products to meet the diverse needs of business and end users.

Designing with New Bus and Device Support

New cross-platform support for PC 97 includes the following:

- Support for ACPI and the OnNow initiative for power management and Plug and Play support, as described in the “OnNow and WDM for PC 97” chapter.
- Support for the new Win32 Driver Model (WDM), with binary compatibility for new drivers across both operating-system platforms, as described in the “OnNow and WDM for PC 97” chapter.
- Support for new universal serial bus (USB) and IEEE 1394 (sometimes called “firewire”).

USB is a serial protocol that runs at 1–12 megabits per second, making it an appropriate bus for input devices, modems, digital audio, and digital image input devices.

IEEE 1394 is a serial protocol that runs at speeds ranging from 100–400 megabits per second, depending upon the implementation. Devices that are prime candidates for IEEE 1394 include digital camcorders and VCRs, digital audio amplifiers, and video teleconferencing.

All new buses supported for PC 97 will support Plug and Play functionality and will include WDM system drivers for both Windows 95 and Windows NT.

- Support in both operating systems for new hardware, including:
 - DVD.
 - Digital image input devices.
 - Network communications devices such as ISDN and cable modems.
 - Connectivity to consumer-electronic devices such as the home stereo, digital VCR and camcorder, digital satellite, and cable TV.
- Support for new Microsoft APIs, including:
 - DirectDraw™ and Direct3D™ APIs for 2-D and 3-D graphics acceleration.
 - DirectSound™ for sound acceleration.
 - ActiveMovie™ and ActiveVRML™ for full-motion video, MPEG-1 acceleration, and other multimedia streaming for both PC-based and Internet applications.
 - Telephony functions for conferencing, voice mail, and fax.
 - Support for multiple monitors, large monitors, and higher quality video display.

SIPC and Designing for Ease of Use

The installed base of personal computers in U.S. homes is still around 30 percent, which is far lower than for any other consumer-electronics device such as TV, VCR, or telephone. A chief barrier to more widespread adoption of the PC in the home is a common feeling among consumers that PCs are difficult to learn, use, and maintain. In addition, many people worry about the perceived short lifetime of a computer and its essential obsolescence within a few years.

To more deeply penetrate the consumer market, the combined elements of the PC design must make it possible for novice users to easily carry out key tasks without experience, including:

- Playing a game or watching a video stream such as a movie or DVD title.
- Writing e-mail or writing a letter for printing.
- Browsing the Internet or listening to voice-mail messages.
- Hooking up another device to the PC.

Research with end users has revealed these areas that call for design attention to broaden the consumer PC market:

- Industrial design for the home, with easy setup and extensibility.
Form-factor issues for such design include size, color, and convenience of connections, switches, and so on. Designing a PC that contains no user-serviceable parts has strong appeal for consumer markets.
The USB and IEEE 1394 external bus architectures will play a key role in many of the ease-of-use scenarios, such as the ability to hot plug any device without rebooting the PC.
- Instantly accessible and quiet.
Key elements of the OnNow initiative serve this design area. The user should be able to expect the PC to perform as instantly as other electronic devices such as VCR or TV, while also expecting the PC to not introduce extra noise into the home environment.
- Sound, video, and other media performance with better quality and better user control than available from other entertainment components.
This means, for example, supporting video display on large monitors with better quality video definition than provided by the TV, or providing PC-controlled audio output at a quality equivalent to home stereos.
- Integration of many separate consumer-electronic devices into one unit.
The PC should be able to control and integrate operation of any standalone device in the home that produces or manipulates information, including the TV, stereo, VCR, CD player, DVD, laser disc, digital satellite services, game console, and so on.

The Simply Interactive Personal Computer (SIPC) initiative at Microsoft is a vision and framework of technologies devised to advance the PC platform by identifying core improvements required to dramatically enhance the level of usability and functionality of both Windows 95 and Windows NT family of operating systems, and PC hardware.

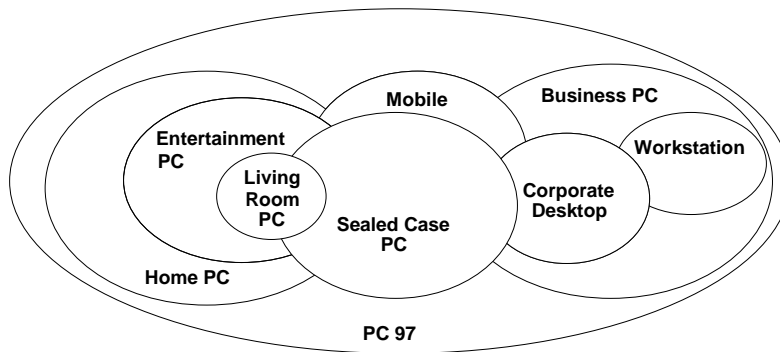
SIPC is not a PC design and the PC 97 guidelines should not be seen as definitions of an SIPC design. The results of the SIPC initiative—both hardware and software—will apply to all categories of systems, including entertainment PCs, mobile PCs, business PCs, workstations, and servers. There is no particular “SIPC” system design, rather just SIPC “focused” hardware technologies, system features, and software designs.

The SIPC initiative can be expressed as a series of goals with specific deliverables associated with them that could apply to any PC hardware platform, form factor, or category of PC platform. SIPC goals also apply to both operating systems, with a high priority at Microsoft to bring the same feature support and functionality to both Windows NT and Window 95.

A good example of a SIPC technology would be OnNow, designed to bring a scaleable set of power management and control features to both PC hardware and the operating system, regardless of the form factor, market category, or platform.

The following diagram represents the technologies that SIPC covers. How they are applied to a market category or form factor is based on the market demands for the product line.

SIPC technologies and PC market categories (Drawing is representational only and is not to scale.)



For PC 97 and SIPC, the term “sealed case PC” is intended to drive the ease of use concept to a new level for the PC platform. Today we have sealed case PCs in the form of mobile and laptop computers. These systems are self contained and provide features to make it exceedingly easy to add and remove components, including memory upgrades, without the end user having to open the PC and partially disassemble boards and cables to add a device to the system.

This concept must be applied in a much broader way to allow the same ease of use for any system form factor. These ideas—that the PC contains “no user serviceable parts” inside, that PCs can be upgraded easily by anyone, and that a sealed-case PC is expandable—are not mutually exclusive, unattainable goals.

The sealed-case PC fits into the SIPC framework of technologies including USB, IEEE 1394, and modular RAM and CPU upgrade. However, Microsoft does not expect that all SIPC designs will be actually “sealed,” even when they do support these expansion methods.

Plug and Play and Designing for Extensibility

A key goal for PC 97 is to provide expansion capabilities “outside the box.” Use of USB and IEEE 1394 are emphasized for connecting devices in PC 97 systems, to ensure that the expansion capabilities are physically accessible to the user and to ensure that the user can take advantage of automatic device configuration and hot plugging capabilities.

A second goal is to manage resources for PC devices to ensure that it is always possible to configure resources for all devices. This means, for example, that any device which uses interrupts always has an interrupt available so that the device can be enabled. The key requirement for meeting this goal is that each bus and device following the relevant Plug and Play specifications.

For 1997, the challenge for Plug and Play is to simplify and unify the underlying mechanisms and deliver a common Plug and Play model for both Windows 95 and Windows NT. The Win32 Driver Model and the Advanced Configuration and Power Interface form the foundation for this common model.

Although VxD-based Plug and Play-compatible drivers will continue to be supported on Windows 95, WDM ensures that future Plug and Play investment is leveraged for both operating systems. In addition to enabling operating system-directed power management, ACPI provides device configuration and generic system event mechanisms for Plug and Play and also provides an operating system-independent interface for configuration control. ACPI leverages the Plug and Play BIOS data structures while providing a processor architecture-independent implementation that is compatible with both Windows 95 and Windows NT.

The previous “Designed for Microsoft Windows 95” logo requirements emphasized Plug and Play as the central focus. For the important buses used in a PC 97 system, the key Plug and Play design work is in place, so that very little new Plug and Play work is designated for PC 97, beyond modifications required on the system board for migration to the ACPI specification.

For PC 97, all devices provided in the system must comply with the current Plug and Play specifications. The operating system must be able to reconfigure the system without requiring the user to restart the PC (with a few exceptions outlined in the detailed requirements in this guide).

Plug and Play for extensibility. A fundamental element of Plug and Play is that each device has a unique identifier. A Plug and Play device identifier is constructed from the logical device identifier structures documented in the specifications for a particular bus, as defined in Part 3 of this guide. The following information is required to create a device identifier for each device on a particular bus:

- **USB:** An identifier that follows the current guidelines defined in the Universal Serial Bus Specification and related device class specification.
- **IEEE 1394:** A global unique 64-bit identifier as defined in the IEEE 1212 Control and Status Register Format Specification.
- **PCI:** A value consisting of the Vendor Identifier, Device Identifier, Class Code, Revision ID, Header Type, Subsystem ID, and Subsystem Vendor ID defined in the Configuration Space header for each PCI device, as defined in the PCI v. 2.1 specification.
- **PCMCIA:** An identifier created from tuples as documented in the “PC Card” chapter in Part 3 of this guide.
- **ISA:** A 32-bit EISA-type identifier for each Plug and Play ISA device. This identifier consists of a leading zero followed by three compressed ASCII characters and a unique 16-bit identifier for each logical device design. To register a new device ID, please send e-mail to pnpid@microsoft.com.
- **Serial (COM) device:** A string of up to 256 characters with five mandatory fields and ten optional fields, as defined in Section 3 of the Plug and Play External COM Device Specification, v. 1.0.
- **Parallel (LPT) device:** A string of up to 20 characters concatenated from MFG, CMD, MDL, CLS, DES, and CID fields, as defined in the Plug and Play Parallel Port Devices Specification, v. 1.0b.

Requirements related to Plug and Play for option ROMs are defined in the “Basic PC 97” chapter in Part 2 of this guide; option ROM requirements for specific devices are defined in the “SCSI” and “ATA and ATAPI” chapters in Part 3, and also in the “Graphics Adapters” chapter in Part 4 of this guide.

Bus choices for extensibility. When choosing expansion bus support for the PC system and determining which devices should use a particular bus, the PC system designer should consider the bandwidth requirements for each particular device, in addition to selecting a bus based on ease-of-use considerations.

For USB, for example, multiple devices will be using the bus. Therefore, as a general guideline, any device that consumes 30 percent or more of the available bandwidth should be implemented using a higher-speed bus such as IEEE 1394. An example for such a choice would be a digital video device such as a digital VCR or camcorder.

A PC 97 system can provide an ISA bus to allow end users to “recycle” their old ISA devices. A PC 97 system that provides ISA support must fully implement Plug and Play boot support, IRQ sharing for ISA (if need to meet resource requirements), and other mechanisms to ensure that an ISA card added to the system can be configured and enabled. Any ISA devices provided with the system must be fully Plug and Play compatible.

Designing for x86-based and RISC-based PCs

Windows NT and Windows 95 are designed to run on platforms that use Intel and equivalent microprocessors, referred to as “x86-based systems” in this guide. These operating systems can run on computers running with Intel 80486-class, Pentium-class, or Pentium Pro-class processors, or equivalent microprocessors that implement the complete 486 instruction set.

Windows NT can also run on RISC-based computers, which includes computers with MIPS R4000 or Digital Alpha 21064 (EV4) or higher processors, or computers that use the IBM PowerPC Architecture.

- When Windows NT is running on an x86-based processor, a virtual-86 processor mode allows direct execution of most instructions in MS-DOS–based applications. A few instructions, such as I/O instructions, must be emulated to virtualize the hardware.
- When Windows NT is running on a RISC-based processor, hardware support for executing x86 instructions is not available, so Windows NT emulates all x86 instructions and provides a virtual hardware environment using the Virtual Device Manager (VDM). The Windows NT VDM also supports ROM BIOS interrupt services, MS-DOS Interrupt 21 services, and virtual hardware for devices using virtual device drivers (VDDs).

This guide provides complete hardware design guidelines for the x86-based computer, whether it is running Windows 95 or Windows NT, plus requirements for RISC-based PCs that run Windows NT. There is no future plan to enable Windows 95 to run on RISC-based PCs.

Advanced RISC computing (ARC) refers to a RISC-based computer architecture standard associated with the ACE consortium. Windows NT runs on top of ARC-compliant PCs, including both RISC- and CISC-based platforms that supply the following components to the operating-system loader:

- For ACE-compliant platforms, such as most MIPS systems, the system firmware supports bootstrap loading and execution as an “abstracted” set of ARC routines and ARC devices.
- For x86-based platforms, a Windows NT hardware recognizer (Ntdetect) finds devices necessary to bootstrap the system by querying the ROM BIOS and then builds a hardware database that emulates the RISC-based ARC firmware.
- For both kinds of platforms, a hardware abstraction layer (HAL) interfaces between the hardware and the system. Device drivers for certain types of devices create an alias between the names of their device objects and the corresponding ARC device name by calling the appropriate API.

In Windows NT, for some devices there are no differences in the requirements for supporting any microprocessor platform. For example, a network adapter driver calls DMA-related functions of the NDIS interface library for DMA operations between the host and the network adapter. These functions support maximized portability, so the driver can run on both x86-based and RISC-based architectures.

However, some differences in microprocessor platform requirements must be addressed in the Windows NT device driver. For example:

- Phase 1 of system startup for Windows NT is microprocessor platform-specific. On x86-based PCs, the hardware boot ROM loads a boot sector, which loads the NTLDR.
For most RISC-based platforms, the firmware loads necessary ARC drivers, acquires hardware configuration data, and loads the OSLOADER.
- Phase 2 of Windows NT system startup sets up memory, captures hardware configuration data, constructs a description of the hardware in memory, and puts a pointer to this description into the loader block.
On x86-based PCs, a hardware recognizer called Ntdetect queries the ROMs, probes the hardware, and collects as much information as possible about the I/O buses and peripheral devices in the PC system.
On RISC-based PCs, the ARC Query Config functions are called to accomplish the same thing.
- On x86-based PCs, there is no difference between a Win32 address and a DMA address as programmed by the MS-DOS–based application to the DMA chip.
On RISC-based PCs, you must add the offset from VDM 0 to DOS 0:0 to convert a DMA address to a Win32 address.

- On x86-based PCs running Windows NT, there are two kinds of video miniport drivers: VGA-compatible miniports and miniports that rely on having the system-supplied VGA miniport driver or another VGA-compatible SVGA miniport driver loaded concurrently.

Miniport drivers on RISC-based platforms rely on the system-supplied VGA support, if it is necessary. For all RISC-based platforms running Windows NT, video miniport drivers need not supply any special support for full-screen MS-DOS-based applications. Instead, video miniport drivers must be set up to configure themselves in the registry with `VgaCompatible=false`.

References for PC 97 Design Issues

This section lists some of the publications, services, and tools available to help build hardware optimized to work with Windows operating systems.

Plug and Play specifications

<http://www.microsoft.com/hwdev/pnpspecs.htm>

Microsoft Device Driver Kits (DDKs) for Windows operating systems

Microsoft Developer Network (MSDN) Professional membership

Advanced Configuration and Power Interface (ACPI) specification

<http://www.teleport.com/~acpi/>

Power management specifications for device and bus classes

<http://www.microsoft.com/hwdev/onnow.htm>

Hardware compatibility tests

<http://www.microsoft.com/hwtest/>

