USB Feature Specification: Interface Power Management

MICROSOFT CORPORATION

INTEL CORPORATION

Revision 1.0rc4c

May 13, 1999

Scope of this Revision

Converted from a review request (rr) format to new format for USB feature specification; no content changes.

Revision History

Revision	Issue Date	Comments
1.0rc4c	5/13/99	Edited Interface Power Management spec; made the changes agreed to at the last Common Class Working Group meeting.
1.0rc4b	3/23/99	Split specification into two specifications: one for interface power management and the other for device power budgeting. This enables the interface power management spec to move along on a faster track than the device power budgeting spec, where there are still many significant issues.
1.0rc4a	2/2/99	Made general changes suggested by Terry Moore (MCCI) and rewrote section 4.3.4, using Terry's ideas.
1.0rc4	1/11/99	Reorganized to make more usable; deleted redundancy; fixed bugs; reformatted prose into tables.
1.0rc3	12/7/98	Incorporated all feedback from Common Class CWG meeting in Tigard, OR
1.0rc2	12/1/98	In section 5.1ff, clarified rule that if a configuration power descriptor is present, there must be a power descriptor for each interface (including alternate interfaces). Rewrote section 4.3. Eliminated obsolete description of "USB System Software Model."Added subsection to section 4.3 that describes how wake-up enabled interface on bus-powered device must resume from USB Suspend. Added minor changes to sections 2 and 3 to make clear scope of this feature is both device and interface power management.
1.0rc1	8/12/98	Expanded Contributors List to include all the people that contributed to the specification. Specified values for new feature selectors in section 6.1 (based on e-mail feedback from jdunn@microsoft.com on 7/13/98). Clarified the rules for including power descriptors on a device in section 5 (based on e-mail feedback from Tonytran@cmdexsvr.cmd.com on 7/10/98).
1.0rc	7/14/98	Accepted all changes in v0.9rd specification to create v1.0rc
0.9rd	6/10/98	Changes suggested by the Common Class Working Group at face-to-face meeting on Long Island, New York, are shown with revision marks in this v0.9rd release of the feature specification
0.9a	1/23/98	Added power consumption and user latency fields to Power Descriptor. Eliminated Appendix
0.8b	1/28/98	Made numerous changes suggested by Common Class Working Group at face-to-face meeting in Atlanta, 1/26/98
0.7	12/2/97	Converted from a review request (rr) format to new format for USB feature specification; no content changes

Contributors

John Dunn	Microsoft Corporation
Mark Williams	Microsoft Corporation
Ken Ray	Microsoft Corporation
Dave Lawrence	GTDI
John Howard	Intel Corporation
Geert Knapen	Philips
Tony Tran	CMD
Dominique Lallement	MGE
Kenny Richards	3Com
Ajay Shah	Phoenix Technologies
Steve Chang	KC Technology
Steve McGowan	Intel
Bruce Ramsland	AMP
Janet Schank	Digital
Mats Webjorn	Universal Access
Chuck Brebanek	Intel
Brad Hosler	Intel

Please send comments via electronic mail to markwi@microsoft.com

For Review and Discussion Only
Draft Document Subject to Revision or Rejection
Not For Publication or General Distribution

Table of Contents

Introduction

1.1 Purpose

This document describes requirements and specifications for providing power management to an interface on a USB device.

Achieving maximum system-wide power savings on PCs is an important design goal in light of the Energy Star™ program, the Five Watt PC goal, and the goal of maximum battery life on mobile computers. With these ambitious low power consumption goals, both PC peripheral and PC platform designers must take power management into account in all aspects of their designs.

USB devices must contribute all they can to system-wide power savings. The *Universal Serial Bus Specification*, which requires all USB devices to implement a low-power Suspend mode (controlled over the USB bus), goes a long way in contributing to system-wide power savings. With the addition of a USB interface power management feature, USB devices can contribute even more power savings.

1.2 Scope

This Common Class feature specification uses the OnNow Power Management initiative architecture and device power management model for PC peripherals. For more information about the OnNow power management model, see section 3.

The information in this document

- Enables USB device designers and developers to participate in a device power management scheme that reduces system-wide power consumption.

USB devices are not required to implement any of the capabilities described in this document.

1.3 Related Documents

USB Specification, Version 1.1, available at http://www.usb.org.
USB Common Class Specification, Version 1.0, available at http://www.usb.org./developers.
USB Device Class Definition for Power Devices, Version 1.0, available at http://www.usb.org/developers.
Advanced Configuration and Power (ACPI) Specification, Version 1.0, available http://www.microsoft.com/hwdev.

1.4 Terms and Abbreviations

Remote wakeup agent	An interface that can be enabled, by the host, to wake up the host when an external event occurs at the interface.
Energy Star™ program	A United States Federal Government program that encourages PC platform makers and monitor makers to produce PCs and monitors that have low power consumption modes. Studies have shown that organizations with a large number of PCs can save significant amounts of money by installing PCs and monitors that have power management features.
Five Watt PC	Design goal set by large PC makers, along with PC component makers. The PC appears off but is quickly ready to use like a VCR or TV. The PC can respond to external events such as a phone ring, arrival of a packet over the LAN, or receipt of signal from a remote control.

2. Management Overview

This section is an overview of the contents of this document and provides a brief summary of each of the subsequent sections. It does not establish any requirements or guidelines.

In general, the *Universal Serial Bus Specification* contains the device requirements that enable the USB host software to contribute to system-wide power management (power savings and wake-up from system sleeping states). These device requirements are:

- A device can draw no more than 500 mA from the USB bus.
- A device can draw no more than 100 mA from the USB bus before being configured by the host software.
- A device can draw no more than 500 μA from the USB bus when it is suspended. The only exception is a high-power devices configured for high-power and enabled as a remote wakeup source, which may draw up to 2.5mA while the device is suspended (for more information, see section 7.2.3 of the *USB Specification, Version 1.1*).
- A suspended device can signal a host that may also be suspended. This notifies the host that it should resume from its suspended mode, if necessary, and service the external event that triggered the suspended USB device to signal the host).

(Note: These bullet items summarize the USB power distribution model given in section 7.2 of the *Universal Serial Bus Specification* and the remote wakeup overview given in section 9.2.5.2 of the *Universal Serial Bus Specification*).

This common class feature specification extends power management to the interface level on a USB device.

Section 3, Functional Characteristics, lists the requirements this interface power management specification must meet, defines all the power states a USB interface can implement, and specifies how backward compatibility with USB interfaces that do not implement this feature is maintained.

Section 4, Operational Model, describes how a device that implements this optional feature interacts with the host and to events on the USB bus.

Section 5, Descriptors, defines a new interface power descriptor and defines the rules of placement for interface power descriptors in a Configuration descriptor for the device.

Section 6, Requests, defines new feature selectors for the Set Feature and Clear Feature requests that an interface must respond to if it implements this optional feature. Section 6 also defines new information returned in response to a Get Status request directed to an interface.

3. Functional Characteristics

This section lists the interface power management feature requirements, defines all the power states a USB interface shall or may implement, and specifies how backward compatibility with USB interfaces that do not implement this feature is maintained.

3.1 Feature Requirements

A USB interface power management feature specification must:

- Define the optional and required power states of a USB interface.
- Maintain backward compatibility with USB interface designs that comply with the *Universal Serial Bus Specification* but do not implement this feature.
- Have a minimal impact on the *Universal Serial Bus Specification.*
- Define the descriptor information that an interface must provide if it implements this feature.
- Define the requests an interface must respond to if it implements this feature.

3.2 Interface Power States

An interface may implement up to four different, distinct power states. A convention for labeling each optional power state is borrowed from the *Advanced Configuration and Power Interface (ACPI) Specification*.

- Each power state is given a unique number, 0 through 3. Intended power savings increase with the increase in the state number.
- Power state numbers are preceded by the letter "D" to indicate transitions between states take place within the device, instead of within the host.

The following table summarizes the meaning of the interface power states D0, D1, D2, and D3.

Table 3-1. D0, D1, D2, and D3 Power State Definitions

Power State	Power Savings at the Interface	Power Consumption at the Interface	Time to Transition Power State to D0
D0	None	Highest	0
D1	More than D0, but less than D2	Less than D0, but more than D2	Not zero, but faster than D2
D2	More than D1, but less than D3	Less than D1, but more than D3	Slower than D1, but faster than D3
D3	Most	Lowest, may be 0	Slowest

3.3 Backward Compatibility with Existing Devices

3.3.1 Host Handling of Devices without Power Descriptors

If the host does not find at least one interface power descriptor in the configuration descriptor that defines the device's current configuration, then the host shall not send any of the Set Feature, Clear Feature, or

Get Status request extensions, defined in Section 6, to the device. For more information about the relationship between host interface power management requests and a USB device, see section 4.1.

3.3.2 Device Handling of Request Extensions

The *USB Core Specification* defines a DEVICE_REMOTE_WAKEUP feature selector for a Device recipient. This specification allows DEVICE_REMOTE_WAKEUP to be used with an Interface recipient.

- A USB device with only one interface can receive either a Set/Clear Feature (DEVICE_REMOTE_WAKEUP) request for a Device recipient or a Set/Clear Feature(DEVICE_REMOTE_WAKEUP) request for an Interface recipient. In both cases, an interface power management-aware device shall enable its one interface for remote wake-up
- A USB device with more than one interface should ignore any Set/Clear Feature (DEVICE_REMOTE_WAKEUP) request for a Device recipient it receives. Such a USB device shall act on any Set/Clear Feature (DEVICE_REMOTE_WAKEUP) request for an Interface it receives if that interface has a wake-up capable bit set in the *bmCapabilities* field of the Interface descriptor.

4. Operational Model

This section describes how a device that implements this optional feature interacts with the host and to events on the USB bus.

4.1 How Device Interfaces Respond to Host Power Management Requests

If it contains one or more interface power descriptors in its firmware, then device logic must be able to respond to the host requests listed in the following table.

Table 4.1

Host Request	Device Response	Comment
Standard Get Configuration request	Return the current Configuration descriptor	This response is already required of all devices in the USB Core specification. However, the Configuration descriptor may contain one or more Interface Power Descriptors.
Extended Get Status request (for a definition of the interface power management extensions to the standard Get Status request, see section 6.2).	Return the current power state (D0, D1, D2, or D3) of the interface and the current remote wake-up state of the interface (Enabled or Disabled).	
Extended Set/Clear Feature requests (for a definition of the interface power management extensions to the standard Set/Clear Feature requests, see section 6.1)	Set the power state of the device to D0, D1, D2, or D3 (as directed by the Set Feature request) and/or set the remote wake-up state of the interface to Enabled or Disabled (as directed by the Set or Clear Feature request).	

4.2 General Device Behavior in the Different Power States

An interface can only receive D0, D1, D2, or D3 requests from the host when:

- The host is in the working state (in other words, the host is not in a power saving mode; such host power-saving modes may be referred to by terms such as Host Standby, Host Suspend, or Host Hibernate).
- The USB hub port to which the device or interface is attached is not suspended.

An interface can only receive D0, D1, D2, or D3 requests from the host when:

- The host is in the working state (in other words, the host is not in a power saving mode; such host power-saving modes may be referred to by terms such as Host Standby, Host Suspend, or Host Hibernate).
- The USB hub port to which the device or interface is attached is not suspended.

In general, the *Universal Serial Bus Specification* contains the device requirements that enable the USB host software to contribute to system-wide power management (power savings and wake-up from system sleeping states). These device requirements are:

- A device can draw no more than 500 mA from the USB bus.
- A device can draw no more than 100 mA from the USB bus before being configured by the host software.
- A device can draw no more than 500 μA from the USB bus when it is suspended. The only exception is a high-power devices configured for high-power and enabled as a remote wakeup source, which may draw up to 2.5mA while the device is suspended (for more information, see section 7.2.3 of the *USB Specification, Version 1.1*).
- A suspended device can signal a host that may also be suspended. This notifies the host that it should resume from its suspended mode, if necessary, and service the external event that triggered the suspended USB device to signal the host).

(Note: These bullet items summarize the USB power distribution model given in section 7.2 of the *Universal Serial Bus Specification* and the remote wakeup overview given in section 9.2.5.2 of the *Universal Serial Bus Specification*).

An interface can only receive D0, D1, D2, or D3 requests from the host when:

- The host is in the working state (in other words, the host is not in a power saving mode; such host power-saving modes may be referred to by terms such as Host Standby, Host Suspend, or Host Hibernate).
- The USB hub port to which the device or interface is attached is not suspended.

After an interface receives a D0, D1, D2, or D3 request and transitions to that power state, it must maintain that state until one of the following events occurs:

- A request to enter a different power state is received (for example, when the device is at D3 and it receives a D0 request).
- A bus-powered device that detects Suspend on the hub port to which it is attached must automatically transition to USB Suspend. A self-powered device that detects Suspend on the hub port should automatically transition to USB Suspend. An exception to this rule is a device that contains an interface that is enabled for remote wake-up (for more information, see section 4.3.4).
- The port to which the device is attached loses power.

In general, when the USB enters USB Suspend, all devices on the bus must go into Suspend (draw 500uA or less from the USB bus). For the interface power management model, this means the total power drawn by the core logic and all the interfaces on the device must be less than 500uA. The only exception to this device behavior is when one or more of the interfaces are capable of remote wakeup, remote wakeup is enabled by the host at one or more of the interfaces, and the interfaces enabled for remote wakeup are at either a D1, D2, or D3 state. In this case, an interface is at a D1, D2, or D3 state following a power state transition request from the host. Following are the device implementation rules for interface power management, USB suspend, and USB resume.

- When the port to which a device is attached is suspended and no interfaces on the device are enabled for wakeup, the core logic and all the interfaces on the device must draw 500uA or less from the USB bus.
- If the port to which a device is attached is suspended and one interface on the device is enabled for wakeup, then the entire device, including the interface enabled for wakeup, must draw 500uA or less from the USB bus.

- A device can be built with more than one interface capable of remote wake-up and each of these interfaces can be enabled for wake-up at the same time by the host. In this case, the entire device, including all the interfaces enabled for wakeup, must be one unit load (100mA) or less.
- If and when a wake-up event occurs at a wake-enabled interface, the awakened interface must automatically remain at the D1, D2, or D3 state. The host must send a D0 request to the awakened interface before the interface transitions to the D0 (fully operational) state.
- When a device that has no interface enabled for wake-up resumes from a USB Suspend, all interfaces on the device must automatically transition to D0, D1, or D2, whichever state the interface was in before the device entered Suspend.

4.3 How Device Interfaces Respond to USB Bus Power Events

The only USB bus physical event that must cause a device to change an interface power state is USB RESET. No other physical events on the bus should cause an interface power state change.

The left-most two columns of the following table lists possible combinations of Interface power states, USB Bus states, and Interface remote-wakeup states. The third column names a Bus event that might occur for a combination and the fourth column shows the required device response to that Bus event. Note that the Device Power Level in the fourth column is the amount of current the entire device can use (power drawn by the core logic plus the sum of the current drawn by all interfaces on the device).

Table 4.2

Interface State; Bus State	Interface Remote Wake-up State	Bus Event	Resulting Power State and Device Power Level
Don’t care; Don’t care	Don’t care	RESET	D0 (Core and all interfaces); 1 Unit max. The device is unconfigured.
Don’t care; Operational	Disabled for all interfaces	Suspend	Same state; 500 uA max
Any interface at D0; Operational	Don’t care	Suspend	Same state; a high-power devices configured for high-power and enabled as a remote wakeup source, which may draw up to 2.5mA while the device is suspended.
All interfaces in either D1, D2, or D3; Operational	Enabled for one or more interfaces	Suspend	Same state; 1 Unit max
All interfaces in either D1, D2, or D3; Suspended	Enabled for one or more interfaces	Interface does remote wake-up	Same state; 1 Unit max

Don’t care; Suspended	Don’t care	Resume	Same state; Appropriate power level for state

5. Descriptors

This section defines the interface power descriptor.

A USB device may implement (a) no interface power descriptor or (b) an interface power descriptor any interface in a configuration (including alternate interfaces).

5.1 Interface Power Descriptor

The purpose of an interface power descriptor is to report

- The power states implemented by the interface.
- The ability of the interface to notify the host of external events.

5.1.1 Interface Power Descriptor Placement

An interface power descriptor must be returned after the interface descriptor for the interface to which the interface power descriptor applies, before any endpoint descriptors for that interface, and before the next interface descriptor (if any). If the host software does not find an interface power descriptor, it treats the device as a legacy USB device as defined in the *Universal Serial Bus Specification.*

5.1.2 Interface Power Descriptor Format

The specification of the interface power descriptor follows.

Offset	Field	Size	Value	Description
0	*BLength*	1	Number	Size of this descriptor in bytes. The value of this field must be 3
1	*BDescriptorType*	1	Constant	INTERFACE_POWER descriptor type (0x08)
2	*BmCapabilitiesFlags*	1	Bitmap	Power state and remote wake-up capabilities of the interface. Bit 0: D0 requests 0 Does not support D0 requests 1 Receives D0 requests Bit 1: D1 requests 0 Does not support D1 requests 1 Receives D1 requests Bit 2: D2 requests 0 Does not support D2 requests 1 Receives D2 requests Bit 3: D3 requests 0 Does not support D3 requests 1 Receives D3 requests Bit 4: Wake-up capability in D1 0 Cannot signal wake-up from D1

				1 Can signal wake-up from D1 Bit 5: Wake-up capability in D2 0 Cannot signal wake-up from D2 1 Can signal wake-up from D2

6. Requests

The *Universal Serial Bus Specification* defines a number of standard requests that all devices must support. This section defines how this feature uses those standard requests, if they differ from the standard implementations. This section defines any additional requests defined for this feature.

6.1 Set Feature and Clear Feature Selectors

Expanding the feature selectors enables the host to set each interface's power state and to set each interface's remote wake-enable state (enabled or disabled). The additional feature selectors required by the interface power management feature are described in the following table.

Feature Selector	Recipient	Description
DEVICE_REMOTE_WAKEUP	Interface	Enables the interface as a remote wake-up source when used with Set Feature; disables the interface as a remote wake-up source when used with Clear Feature.
INTERFACE_POWER_D0	Interface	Interface transitions to D0; used only with Set Feature.
INTERFACE_POWER_D1	Interface	Interface transitions to D1; used only with Set Feature.
INTERFACE_POWER_D2	Interface	Interface transitions to D2; used only with Set Feature.
INTERFACE_POWER_D3	Interface	Interface transitions to D3; used only with Set Feature.

The values for the new feature selectors are given in the following table.

Feature Selector	Recipient	Value
DEVICE_REMOTE_WAKEUP	Interface	1
INTERFACE_POWER_D0	Interface	2
INTERFACE_POWER_D1	Interface	3
INTERFACE_POWER_D2	Interface	4
INTERFACE_POWER_D3	Interface	5

6.2 Get Status Request Extensions

The information returned by a Get Status request to an interface is expanded to include the current power state of the interface and the current wake-enable status of the interface. This enables the host software to determine the current power state and the current wake-enable state of any interface at any time.

D7	D6	D5	D4	D3	D2	D1	D0
Reserved (reset to zero)				Current power state (0, 1, 2, or 3)		Remote wake-up state	Reserved (reset to zero)
D15	D14	D13	D12	D11	D10	D9	D8
Reserved (reset to zero)							

Bit D1 indicates the remote wake-up state:

0 Disabled
1 Enabled

Bits D3 and D2 indicate the current power state:

00 Power state D0
01 Power state D1
10 Power state D2
11 Power state D3