USB Mobile System Design Guidelines

Revision 1.0
11/06/96

Table of Contents

Revision History

Revision Number	Date	Author	Notes
1.0	11/6/96	R. Dunstan	Release for Publication

1. INTRODUCTION

The Universal Serial Bus (USB) is a communications architecture that is designed to allow a user to easily connect peripheral devices to a computer using a common cable/connector. USB uses a 4-wire cable; two of the wires provide a 12Mbs serial communications link, while the remaining pair of wires supply power to the downstream device.

USB when applied to mobile systems poses a number of design choices to the system designer. Identifying these choices and implications will help implementers design robust mobile USB systems.

1.1 Scope

This document focuses on hardware related issues with incorporating USB into mobile systems. Each issue is then explained and possible solutions are offered. Mobile-related extensions, changes and/or clarifications to the USB core specification and/or the USB Device Class specifications are under discussion, but alternatives provided in this document allow implementation with the present specifications. While the focus is on application of USB in mobile systems, many of the same issues and solutions may be applied to desktop systems as well.

1.2 Audience

The intended audience for this document includes OEM's implementing USB in mobile systems.

1.3 Relationship to core specifications

This guideline restricts and/or expands the device and host requirements stated in the USB Core Specification, based on mobile requirements.

Whereas the USB Core Specification describes power from both the upstream ports' and the devices' point of view, this guideline focuses on the upstream port (power source). This document refers to power (watts), while the core specification refers to current (amperes). This is to relate the USB's requirements on the mobile system's power supply output current which is fixed at 5 volts. The relationship between power and current is that power equals current times volts.

2. REFERENCES

The USB specifications can be found on the USB website *http://www.teleport.com/~usb*.

Universal Serial Bus Specification, v1.0 1996

Universal Host Controller Interface Design Guide, Revision 1.1, March 1996

PC97 Draft, Revision 0.7, March 1996, Microsoft

3. DEFINITIONS

- **ACPI:** Advanced Configuration and Power Interface, V0.6, Apr 1996
- **APM:** Advanced Power Management. A BIOS interface defined to enable system-wide power management control via software.
- **Bus-Powered Hub:** A USB hub that derives all its power for both internal functions and for downstream ports from its upstream connection. This type of hub will commonly be packaged with another USB device such as a keyboard. A bus-powered hub is a high power device.
- **Downstream:** Farther from the Host than the present location.
- **Downstream port(s):** A port that is electrically connected to the hub, but is farther away from the Host than the root port.
- **Five unit load:** A five unit load is five times the one unit load or less than or equal to 500 mA or 2.5 watts from the 5 volts supplied by the USB connector.
- **High-Power Device:** A USB device that requires up to 5 unit loads (500 mA) from the USB connector to provide full functionality. Note: It must draw no more than one unit load upon power up (100 mA), but it may draw up to five unit loads (500 mA) after configuration.
- **Low-Power Device:** A USB device that requires no more than one unit load (100 mA) from the USB connector to provide full functionality. Note: It must draw no more than one unit load upon both power up and configuration.
- **Mobile Systems:** Computer systems that are designed for mobile service and are capable of being battery operated. Notebook computers and PDAs are two examples of mobile systems.
- **One unit load:** A unit load is defined as being less than or equal to 100 mA or 0.5 watts from the 5 volts supplied by the USB connector.
- **Port:** The attachment point for a USB device.
- **Port state:** The logical state of a port. The possible states include: ENABLED, DISABLED and SUSPENDED. In some cases, the state may also refer to power availability at the port (e.g. on or off).
- **Remote Wakeup:** The ability of a Suspended device to cause a suspended system to wakeup on some event. For example, a modem that wakes up the system when it detects a ring.
- **Root Port**: The port on a device which connects the device to the Host. Only hubs implement more ports than the mandatory root port. Also known as an upstream port.
- **Root Hub Ports**: Ports on the back (or front) of the mobile system.
- **SOF:** Start of Frame. SOFs are used to demarcate the frames on the bus and are used as an indicator of bus activity.
- **UHCI:** The Universal Host Controller Interface.
- **Upstream:** Closer to the Host than the present location.
- **Upstream port(s):** see Root Port.
- **USB:** Universal Serial Bus
- **USB Bus Segment:** The physical connection between a port and the attached USB device.

4. BACKGROUND

This section is intended to provide the reader with some background about USB. The design philosophy, architecture and power features of USB which may affect mobile systems are covered. The reader is particularly encouraged to read sections 4.1 and 4.7 to get a high level overview of how USB affects the mobile system. Sections 4.2 through 4.6 describe in some detail how USB operates and provides important background information to understand how USB can affect power consumption. This section is intended to provide the reader with sufficient background to understand the areas where USB affects mobile design. It is *not* intended to replace a thorough reading and understanding of the USB specifications.

4.1 Mobile USB Design Philosophy

The core USB Specification specifies power requirements that are very appropriate for an environment where USB power requirements represent a very small fraction of the overall system power budget. In such environments, the existing system power supply can readily handle the additional instantaneous and steady-state power requirements of USB device attachment and detachment. USB devices assume that they can immediately draw the power level which is legal for their present state.

In contrast, a battery-powered mobile system typically has a very limited power-supply capacity which is carefully apportioned to meet the system's power requirements. The Host (rather than individual devices) controls power allocation. Device power is controlled so as to minimize large step loads on the power supply. System-level power management controls power in accordance with the system's power policy by adjusting a device's power state while taking the device's latency and the system's ability to supply power into account. In short, the Host makes all decisions with respect to when a device may take power and the Host controls the rate at which a device is allowed to consume power.

When implementing USB in a mobile system, a balance has to be reached between the power requirements of USB and the power constraints of the mobile system. Should the system power supply be increased or should some power/function tradeoffs be made? Can some of the tradeoffs be offset by a docking station or an external self-powered hub? In the final analysis, implementing full desktop-equivalence must be balanced against the possible increased size and cost of the power supply and/or shortened battery life.

4.2 System Architecture

USB uses a star/hub architecture. USB devices and hubs are connected downstream from the Root Hub on the Host, forming a tree like structure with sub-trees. A USB device provides a function, such as a mouse, keyboard, modem or hub. Only hubs provide connection points for other USB devices.

The Root Hub is the highest level of connection to the Host. The USB specification requires one (1) Root Hub, connected directly to the host controller. As with the host controller itself, some part of the implementation of the Root Hub will typically be a software driver, rather than just a hardware component. A mobile system will generally have only one USB port, however additional ports may be present if desired. The USB Host implementation consists of a USB host controller (usually packaged with a Root Hub) plus additional software in the operating system.

Figure 1 below shows one possible USB topology. The key features inside the mobile system (rectangle) are: a Host, a host controller, a Root Hub and a Root Hub port. External to the mobile system are: hubs and devices. Note that a hub is also a device.

USB impacts many areas of a mobile system, ranging from the chipset to the software stack that presents USB devices to the operating system. Figure 2 presents that hierarchy.

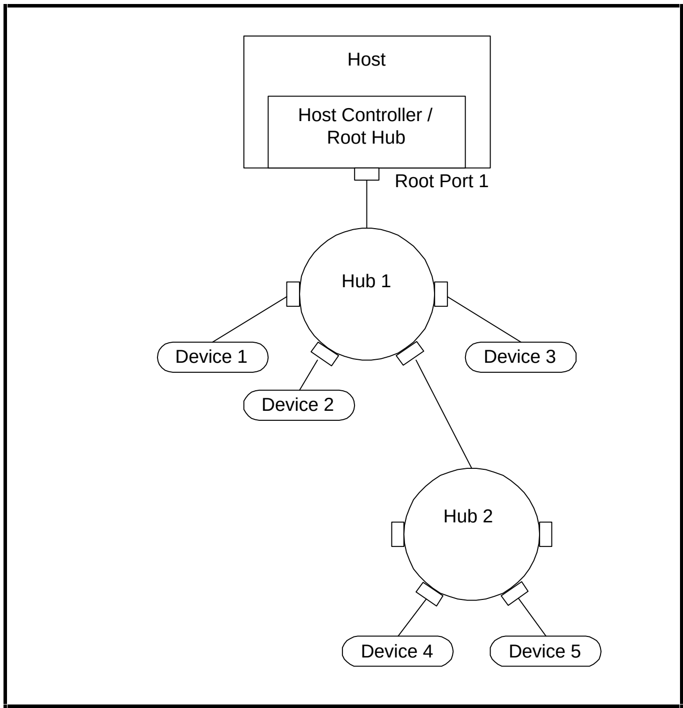

Figure 1: Sample USB Topology

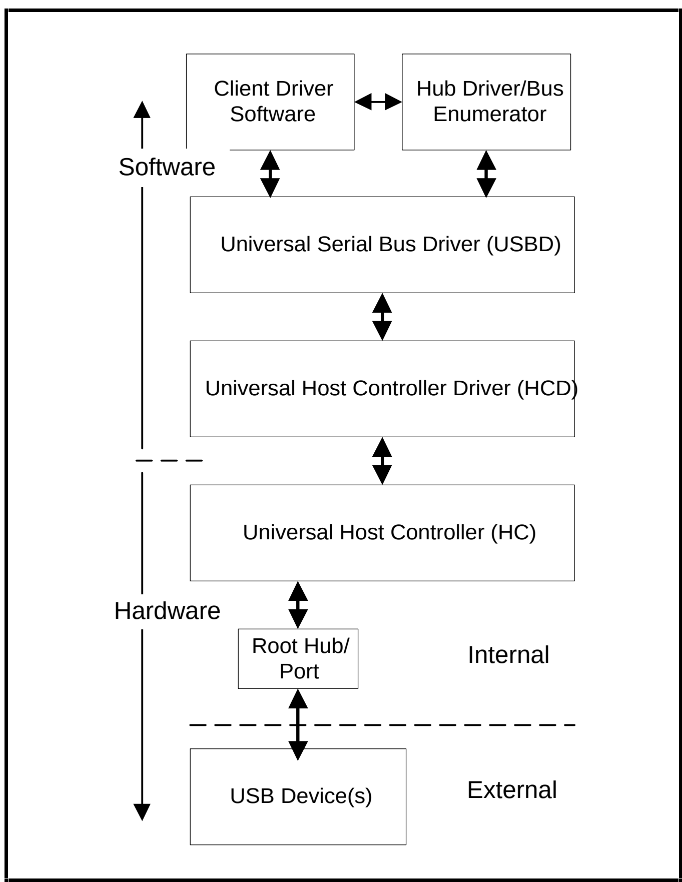

Figure 2: USB Hardware/Software Hierarchy

Figure 2 shows the major parts of the USB implementation.

The Client Driver software may be provided by the OS vendor or another 3rd party vendor for standard USB device classes. For example, there are device classes for printers, human interface devices etc. If a vendor supplies a USB device that does not conform to a class specification, they will also have to supply the appropriate driver software with the device.

A Hub Driver/Bus Enumerator will be provided by the OS vendor or another 3rd party vendor. It controls ALL hubs in the USB system INCLUDING the Root Hub. It is an integral part of the USB power policy, determining when a bus segment or device can be suspended or resumed. It may be replaced by an OEM who desires to differentiate their system's performance and power characteristics.

The USBD is provided by the OS vendor. It is the central driver to USB. It provides the bus scheduling; for example allowing a device to get a guaranteed amount of the bus bandwidth. It is also responsible for managing data transfers over the bus and other bus housekeeping duties.

The HCD is provided by the OS vendor. It hides details of the hardware implementation from the USBD. Currently, there are two hardware interfaces supported, the UHCI (Universal Host Controller Interface) and the OHCI (Open Host Controller Interface).

4.3 USB Device Power relative to a USB bus segment power state

For USB, the state of a port determines the state of the attached USB bus segment which in turn determines the device's power consumption and behavior. This means that a USB device's power is NOT always directly controlled, rather it is a side-effect of the state of its upstream port(s). There is one exception, in which the transition between low and high power is controlled by the configuration of the device. Table 1 describes how a USB device responds to a change in an upstream port's state. Note that the configured power draw is less than or equal to the maximum power consumption reported by the device for its configured state, which in turn is limited by whether it is a high or low power device. For a more detailed state transition diagram, refer to the USB specification Figure 9-1.

Old State	Transition	New State	Comments
Port: DISABLED *Device*: Suspended *Power*: 500 µA max	*Action*: SetPortFeature(PORT_ENABLE) *Signaling*: none	*Port*: ENABLED *Device*: previous state *Power*: configured	Hosts should never enable ports other than by resetting them.
Port: DISABLED *Device*: Suspended *Power*: 500 µA max	*Action*: SetPortFeature(PORT_RESET) *Signaling*: hub/port generates RESET signaling for the specified time.	*Port*: ENABLED *Device*: Default *Power*: 100 mA max	
Port: SUSPENDED *Device*: Suspended *Power*: 500 µA max	*Action*: ClearPortFeature(PORT_SUSPEND) *Signaling*: hub/port generates RESUME signaling for the specified time.	*Port*: ENABLED, not SUSPENDED *Device*: previous state *Power*: configured	The Host must explicitly resume a SUSPENDED port, which resumes only the device immediately connected to that port. A device issuing a Remote Wakeup elsewhere on the bus does not resume SUSPENDED ports. A resumed device is allowed to transition immediately from 500µA max to its configured power (up to 500 mA).
Port: SUSPENDED *Device*: Suspended *Power*: 500 µA max	*Action*: RESUME signaling from attached device. *Signaling*: hub/port generates RESUME signaling to all ENABLED ports for the specified time.	*Port*: ENABLED, not SUSPENDED *Device*: previous state *Power*: configured	All ENABLED ports, including the Root Port will be resumed.
Port: ENABLED *Device*: as configured *Power*: 100 mA up to 500 mA, based on configuration	*Action*: none *Signaling*: normal traffic	*Port*: ENABLED *Device*: as configured *Power*: 100 mA up to 500 mA, based on configuration	All segments between the device and the Host are active.
Port: ENABLED *Device*: Suspended *Power*: 500 µA max	*Action*: none *Signaling*: no traffic	*Port*: ENABLED *Device*: Suspended *Power*: 500 µA max	At least one port between the device and the Host is SUSPENDED. However, Remote Wakeup's issued by other devices DO wake up Suspended devices attached to ENABLED ports. See SUSPEND entries.

Table 1: Device Response to Upstream Power-State Transition

Table 2 summarizes the maximum allowable power consumption from an upstream connection by a USB device. Note that a bus-powered hub is a high-power device that is an exception to the suspend current specification and can consume up to 2.5 mA from its upstream port (i.e. 500 µA for each device attached to the hub plus 500 µA for the hub itself). Note that, in a compound device, additional devices may be embedded with the hub. In that case, the embedded device is power budgeted independently by the Host from the hub. Hubs may also be self-powered devices and then must obey the rules given below.

	Suspend Current	Reset Current (From Attached until Configured)	Maximum Configured Current
Low Power Device	≤ 500 μA	≤ 100 mA	≤ 100 mA
High Power Device	≤ 500 μA	≤ 100 mA	≤ 500 mA
Self Powered Device	≤ 500 μA	≤ 100 mA	≤ 100 mA
Bus Powered Hub	≤ 2.5 mA	≤ 100 mA	≤ 500 mA

Table 2: Allowable Device Power Consumption from Upstream Source

4.4 USB Suspend and Resume

Systems may suspend all, or any sub-tree, of an instance of USB. To globally suspend the USB system, the USB controller stops sending SOF packets. The lack of bus traffic causes each attached USB device to go into its suspend state.

The USB system, or any of its sub-trees, may be resumed in one of two ways:

- The Host can explicitly resume a single port (including the ports on the Root Hub), which results in RESUME signaling to the downstream device, after which the device will again see SOFs. If the device is a hub, any device attached to ENABLED ports on that hub will also see SOFs and so also wake up. This process will continue until all sub-trees with devices attached to ENABLED ports of resumed hubs are awakened.
- A USB device may change the DC state of the USB signal connections and cause a Remote Wakeup. This wakes up the hub immediately upstream from the device, and eventually, the Host. It will also wake up all devices/hubs attached to ENABLED ports on any hub which is on the path between the Host and the device issuing the Remote Wakeup.

In both cases ALL the resumed USB devices begin consuming power at their former rate. In the second case, they can start consuming power BEFORE the Host is notified that the wakeup event has occurred.

Remote Wakeup can cause the attached devices to return to their former power state prior to the host controller receiving notification of the resume event. In the case of a typical mobile implementation with only one USB port, the mobile system's power supply will be required to supply 5 volts at 500 μA (2.5 mA for a high power port) when suspended, and then without notification, supply up to 100 mA (500 mA for a high power port) to satisfy the potential active USB device load.

While this will work in mobile systems, such an unrestricted resume's effect on system power consumption is not optimal. Selectively suspending devices (i.e. "pruning" the tree prior to the system suspend), however, can reduce the potential impact on the system power supply when the system is resumed. The following section describes how to manage selective suspend and its interactions with the hub, port, Host and device.

4.5 Selective USB Suspend and Resume

The behavior of the hub ports with respect to Suspend, Resume, and Enable/Disable are key to device power consumption and power control. As a result, a good understanding of how selective suspend and resume work in USB is vital to its application in mobile systems.

Selective suspend is a method which relies on the ability to suspend a specific USB port. Any USB device(s) downstream of that port then also go into suspend. Therefore, it is possible to suspend an entire branch of a USB system simply by suspending its upstream port.

Selective suspend depends on the system's ability to control a hub's ports. A port may be ENABLED or DISABLED. An ENABLED port may be SUSPENDED or not. The following explains in detail the states that a port may be in and the effect on downstream USB devices.

4.5.1 Port Disable

A port may be disabled by a command to the hub or due to an error condition (babble) detected by the hub. All devices downstream of a DISABLED port will suspend (≤500 µA) because the port will no longer broadcast SOF messages to those devices. When the port is in the DISABLED state, it must be explicitly put in the ENABLED state by the Host before any downstream device(s) can be accessed.

The best method to enable a port is to Reset it, which results in RESET signaling to the device and an ENABLED port.

4.5.2 Port Enable

When the port is in the DISABLED state, it must be explicitly put in the ENABLED state by the Host. This should be accomplished by resetting the port. After a reset the device is in the Default state and is limited to 100 mA until it is re-configured.

If the port is ENABLED without being reset, the device returns to its previous state and its configured power draw (up to 500 mA). Due to power considerations and device state management it is STRONGLY recommended that the Host only transition ports from DISABLED to ENABLED through the use of Reset.

4.5.3 Port Suspend

USB devices are connected to a bus segment. Whenever the bus segment to which a USB device is connected is suspended, by the lack of bus traffic, the device must suspend its operation and reduce its power consumption from the bus to less than 500 µA. USB devices themselves DO NOT have a suspend command, rather a power state based on traffic on the bus segment to which they are attached. This means that to explicitly suspend a USB device, the device's upstream hub's port must be controlled by the system software.

All USB devices, including hubs, are required to go into suspend when they do not see traffic (Start of Frame messages) on the bus for some period of time. This may result from one or more of the following conditions. Refer to Figure 3 for examples of the following conditions:

1. The port to which the device is attached has been disabled by a command to the hub or due to an error condition (babble) detected by the hub. (Hub 1 detected a babble condition from Device 3 and Disabled Port 4.)
2. The port to which the device is attached has been selectively suspended by the Host. (Host selectively Suspended Hub 1 : Port2, causing Device 2 to suspend.)
3. Some port above the device's root port has been selectively suspended by the Host, but the device's root port is still ENABLED. (Device 5 is Suspended because Hub 1 : Port 3 is Suspended, so no SOFs are present downstream.)
4. The host controller has stopped generating all traffic (e.g. no SOFs are sent downstream), probably due to a system power change.

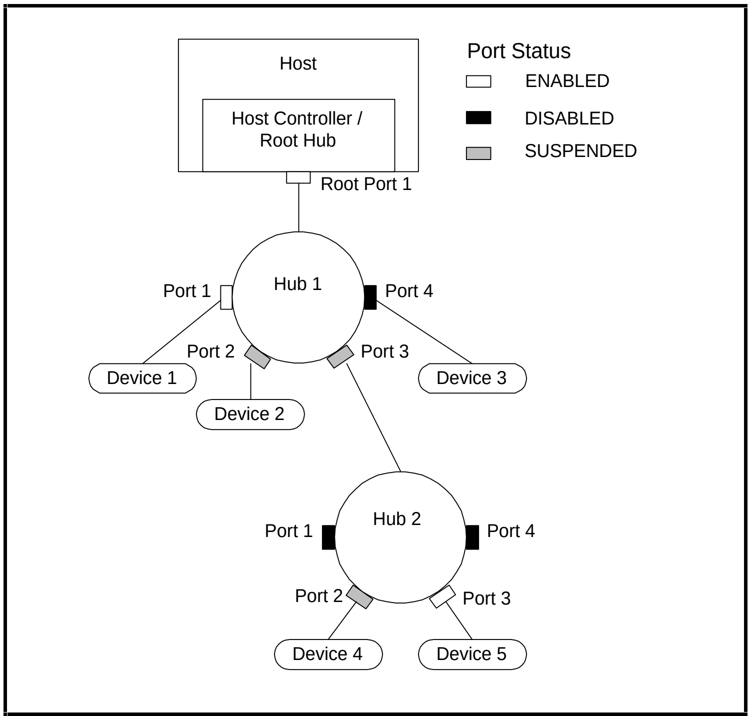

Figure 3: USB Device Suspend/Resume Sample System

Table 3 summarizes the USB devices' and hubs' power states for the USB system previously described in Figure 3. The top row (shaded) indicates the situation when the Host is active. The remaining rows address various situations when the Host is suspended. The table indicates which devices are capable of issuing a Remote Wakeup and summarizes which devices are resumed when a particular USB device receives a wakeup event.

	Hub 1	Hub 2	Device 1	Device 2	Device 3	Device 4	Device 5
Host active	active	Suspended	active	Suspended	Suspended	Suspended	Suspended
Remote Wakeup capable?	yes	yes	yes	yes	no	yes	yes
Device 1 does Remote Wakeup	resumes	Suspended	wakeup/ resume	Suspended	Suspended	Suspended	Suspended
Device 2 does Remote Wakeup	resumes	Suspended	resumes	wakeup/ resume	Suspended	Suspended	Suspended
Device 4 does Remote Wakeup	resumes	resumes	resumes	Suspended	Suspended	wakeup/ resume	resumes

Table 3: USB Device Power States for Suspend/Resume Sample System

4.5.4 Resume

The actions taken upon RESUME signaling are dependent upon the state a hub port is in and dependent upon where the Resume is initiated. Each case of port state will be examined.

4.5.4.1 Upstream port DISABLED

When the port is DISABLED, it must be explicitly enabled by the Host. If this is done using Reset, this also causes the attached device to reset to the Default state. Although the device may have actually been in a Suspended state, the device below a DISABLED port is in the attached state from the Host point of view and will never be resumed.

In Figure 3, Device 3 is attached to Hub 1 : Port 4 which is DISABLED. In order for the system to access the device, Hub 1 : Port 4 must be enabled. At that time, Device 3 is in the Default state awaiting configuration and is consuming up to 100 mA.

4.5.4.2 Attached port Selectively SUSPENDED

The Host initiates RESUME signaling by resuming a SUSPENDED port on a particular hub. When the Host resumes a port on a hub, the hub sends resume signaling down just that port for a specified period and then leaves the port in the ENABLED state. If the host controller has been suspended, the bus is resumed by resuming all of the ports on the Root Hub. When a SUSPENDED port is selectively resumed by the Host, the attached Suspended device is awakened by RESUME signaling generated by the hub followed by bus traffic. The device resumes power consumption to its previous power state which may have been up to 500 mA.

A Suspended device initiates a resume by driving RESUME signaling up its root port. In order for this RESUME signaling to reach the upstream hub, the port to which the device is attached must not be in the DISABLED state. The port must be in either the ENABLED or the SUSPENDED state.

Regardless of who initiates the resume, a hub reflects the resume signaling it receives into all of its ENABLED ports. SUSPENDED and DISABLED ports are not resumed. Whenever the hub sees resume signaling on any of its ports, including its root port, it drives resume signaling down the rest of its ENABLED ports, including its root port. The resume signaling is propagated upward until it reaches a hub which is not Suspended. That hub interprets the resume event as a status change, and sends it to the Host.

When the attached device has its Remote Wakeup feature enabled and does a Remote Wakeup, the device resumes at its previous power state.

In Figure 3, Device 2 may be resumed by the Host controller resuming Hub 1: Port 2. If it has Remote Wakeup capabilities, and those capabilities are enabled, then if a Remote Wakeup event occurs the device will resume and signal Hub 1, then Hub 1 will report a status change to the Host.

4.5.4.3 Upstream port DISABLED

When any port is SUSPENDED, then all devices downstream of that port are Suspended. As in the previous example, either the Host or the device may initiate the resume.

In Figure 3, Device 5 is on an ENABLED port, BUT Hub 2 is in a Suspended state because it is attached to a SUSPENDED port, Hub 1 : Port 3. If Device 4 has Remote Wakeup capabilities which are enabled and has a wakeup event, it signals a resume to Hub 2 : Port 2. Hub 2 reflects the resume signaling to Port 3 and to its root port. Hub 1 : Port 3, receives the Remote Wakeup event and reports it to the Host as a status change. Device 4 and Device 5 resume in their previous power state. Hub 2 also returns to its previous power state. This means that the Host's power supply must be able to supply at least 1.5 watts BEFORE the Host receives notification of the wakeup event.

4.5.4.4 Selectively Suspending Root Hub Ports

As in the previous case, all devices below the port on the Root Hub are Suspended thus effectively suspending the entire bus. RESUME signaling can be either from the Host or a device. On a subsequent resume, incremental power consumption may be an issue.

A Host can have a policy to limit the power draw on resume. In this case, the Host will selectively suspend ports so as to prune the bus tree, leaving only path(s) to those device(s) it wishes to have resume automatically, either when the Host resumes an upstream port of that device, or when another device issues a Remote Wakeup . It must also disable Remote Wakeup for those devices which it does not wish to cause a resume. A mobile system may also implement a policy to turn off power to the USB whenever it suspends while operating on its battery.

DISABLED ports prevent upstream AND downstream traffic and usually result in a downstream reset when they are enabled. SUSPENDED ports prevent downstream traffic and result in downstream RESUME signaling when they are resumed.

Figure 4. describes how RESUME signaling is propagated through a hub and illustrates how it is dependent on its individual port states. RESUME signaling can be initiated by the Host or by a device. Host initiated RESUME signaling goes downstream through all ENABLED ports. Device initiated RESUME signaling goes upstream and is reflected to all ENABLED ports.

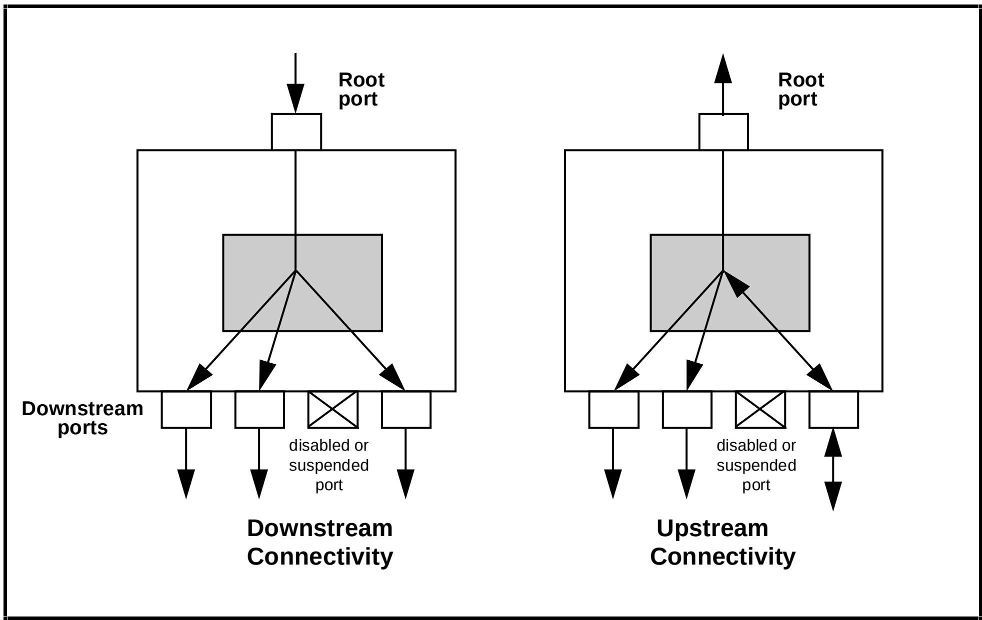

Figure 4: Resume-Signal Propagation Through Hub

4.6 USB Device Attach/Detach

When a USB device is first attached to a port, it consumes power before there is any notification to the Host. The in-rush current is specified as that which will flow into a 10 µf capacitor in parallel with a 44 ohm resistor. The exact device power consumption characteristics prior to reset and receipt of SOFs is not specified. A simplifying assumption is that it is a constant current load, not to exceed 100 mA.

There is a step decrease in the load when a USB device is detached (unplugged). The decrease may be as much as 500 mA and is dependent on the USB device's present configuration. As with attach, the load is removed before the Host is notified of the event.

4.7 Summary of USB Device Behavior

The key characteristics related to device power and power control are:

1. A USB device is allowed to take 100 mA immediately after it is attached, until it is configured.
2. A USB device may drop as much as 500 mA load from the USB supply when detached.
3. A USB device can consume up to 500 mA in some configurations.

   Note: The configuration of a USB device is under Host control and the Host may choose how and if a device is configured based on the Host's power policy.
4. A USB device's power state is controlled by the state of the bus segment upon which it resides. If any port upstream of the bus segment is SUSPENDED or DISABLED, then that bus segment is suspended (no SOFs present). The power consumption is at most 500 µA when the bus segment is suspended.
5. When a USB device does a Remote Wakeup, power can be consumed from the Host before the Host is notified. All devices attached to enabled bus segments also receive the resume message and can consume power prior to the Host's notification.
6. A USB device resumes to its last power level. If a USB device was configured to take the maximum allowable 500 mA at the time it was suspended, then it will resume at that power level.

5. USB IN MOBILE SYSTEM ISSUES

5.1 Hardware Related Issues

When USB is included in a mobile system, careful attention must be paid to the USB power source. Since the USB bus is always powered and USB devices take power when attached, the system must be designed to withstand or control the resulting transients on its power supply.

Each of the topics discussed in this section includes "Explanations" of the subject situation, followed by associated design "Recommendations".

5.1.1 Power Supply

USB imposes the following requirements on the mobile system's power supply, including some on the 5 volt supply.

	High-Power Root Hub Ports	Low-Power Root Hub Ports
Attachment Surge Capacity	0.5W @ 5v per root port; 10 µF || 44W inrush	0.5W @ 5v per root port; 10 µF || 44W inrush
Resume Step-Change Capacity	12.5mW to 2.5W, within 20ms	2.5mW to 0.5W, within 20ms
Detachment Step-Change Capacity	2.5W max to 0	0.5W max to 0
Steady-State Capacity	2.5 W (500 mA @ 5v)	0.5 W (100 mA @ 5v)
Minimum Power (Suspend)	12.5 mW per Root Hub port	2.5 mW per Root Hub port
Maximum Power	2.5 W per Root Hub port	0.5 W per Root Hub port

Table 4: USB Power Supply Requirements

USB devices differ from other devices that are connected to mobile systems. Devices connected to the serial, parallel and PS2 ports are usually connected prior to boot and take little or no power from the mobile system. PCMCIA/CardBus devices are inserted/removed dynamically, but their power is controlled at all times by the Host. USB devices can take relatively large amounts of power, prior to Host control. For example:

- The Host must be prepared to supply up to 0.5 watts (100 mA) when a USB device is attached.
- The Host may ultimately be required to supply up to 2.5 watts (500 mA), depending upon device configuration, which is under Host control, AND if the Host has a high power port.
- When a Remote Wakeup event occurs, the Host must be prepared to supply up to 2.5 watts per port BEFORE it is notified that a Remote Wakeup event has occurred.

Although USB devices are supposed conform to the specification and limit their current consumption, the 5-volt USB power supply should additionally be well protected to prevent failures due to misbehaving devices.

The following sub-sections describe several particular situations which entail special mobile power-supply design considerations. Many of the cases below have common recommendations. For example, many of these may be altered by the BIOS setup.

5.1.1.1 No attached devices

Explanation

The mobile system has to supply 5 volts with a capacity of at least 0.5 watts per port (2.5 watts per high power port). Power supply efficiency may be an issue as the power supply needs to operate efficiently over a wide dynamic range from no load, or the minimal load presented by Suspended devices, up to its maximum design capacity.

Recommendations

⇒ Design the power supply so as to minimize the no load power requirements.

⇒ Provide the user with a BIOS setup option that disables USB when the mobile system is battery operated. This has two effects: the USB system can not detect when devices are attached or detached, and the USB system can not respond to device wakeup events. While it will have a minimal impact on battery life, it may eliminate the need for the presence of a 5 volt supply during suspend. Turning off the USB 5 volt supply during suspend prevents devices from consuming 100 mA when attached prior to system notification.

⇒ Use a USB connector inside the notebook that can signal the mobile system when a USB device is connected. The mobile system would use the signal to turn on power to the USB connector.

⇒ Any combination of the above.

5.1.1.2 USB Device Insertion - Step load

Explanation

The mobile system must supply 5 volts at the USB connector. When a USB device is connected it presents an instantaneous load of up to 100 mA to the system. The USB specification requires that the device reach a stable state within 10ms which may place certain stability requirements on the power supply.

Recommendations

⇒ Ensure that the power supply meets the requirements, particularly when battery operated.

⇒ Provide the user with a BIOS setup option that disables USB when the mobile system is battery operated.

⇒ Use a USB connector inside the notebook that can signal the mobile system when a USB device is connected. The mobile system would use the signal to turn on power to the USB connector.

⇒ Use sufficient bulk capacitance (e.g. 200 - 250 µf) in the power supply or at the USB port to absorb the initial surge. This issue is presently under discussion in the USB Device Working Group.

⇒ Allow the power supply to start and become stable over a period of time greater than 10ms.

⇒ Any combination of the above.

5.1.1.3 Ports on Mobile Root Hubs Supporting only low power devices

Explanation

The mobile system must be able to supply 5 volts at 500 mA at the USB connector in order to meet the USB requirement that ports on the Root Hub are able to supply 1 high power load (2.5W). This can require a bigger battery and/or decreased battery life and/or increased cost of the AC adapter (e.g. brick).

Recommendations

⇒ Design the port on the Root Hub to support only low power loads. This may be a reasonable alternative for systems that can not tolerate the additional load. A self-powered hub can be used to connect high-power devices. Eliminating support for high power devices may preclude the use of some interesting peripherals which require a high power load (e.g. a fully functional IDSN terminal adapter).

⇒ Support only low power loads when operating from the battery, allow high power loads when operating from AC.
⇒ Provide the user with a BIOS setup option that selects the USB power levels that the mobile system will support when operating from the battery and/or AC.

5.1.1.4 Suspend time

Explanation

The Host must supply each USB port on the Root Hub with 12.5mw (500 µA) when suspended. Typical notebooks have less than a 20mw power budget. Thus, USB requirements must be taken into consideration or suspend time will be reduced.

Recommendations

⇒ Ensure that the power supply is very efficient at low loads, particularly when battery operated.
⇒ Provide the user with a BIOS option to turn off USB when suspended while battery operated. This should include turning off power to any internal USB devices. Note: implementing internal USB devices is not recommended due to the increased power requirements of a USB device interface (e.g. SIE) over other internal alternatives.

5.1.1.5 Remote Wakeup - load presented to power supply

Explanation

The mobile system must have the ability to supply 5 volts at up to 500 mA at each USB port on the Root Hub while suspended in order to meet the potential load that can occur when a Remote Wakeup is issued. The power can be consumed BEFORE the Host receives the wakeup notification. Each USB port on the Root Hub must be able to transition from its suspend power level to potentially 2.5 watts when a wakeup event occurs. The new power level must be stable within 10 milliseconds.

All devices connected through ENABLED ports back to the Host will respond to the Remote Wakeup by consuming power at their pre-Suspended rate even before the Host receives the wakeup event.

Recommendations

⇒ Ensure that the power supply meets the requirements, particularly when battery operated.
⇒ Provide the user with a BIOS setup option that disables USB while battery operated.
⇒ Provide the user with a BIOS option to disable USB when suspended while battery operated.
⇒ Make sure the USB hub driver “prunes” the USB tree to minimize the number of devices that receive the wakeup event. Note: The hub driver may be provided by the OS vendor, the hub manufacturer or the OEM. Hub drivers are central to the USB power management strategy and may be made power management aware without necessarily incurring changes to the rest of the USB stack.
⇒ Sufficient bulk capacitance in the power supply will assist with the onset of USB current when the mobile system resumes its normal operation.

5.1.2 Reverse Power Leakage

Explanation

While USB supplies power downstream, a situation arises when the upstream side is powered off and the downstream device is self-powered. In this case, the USB specification is unclear as to where the pull-up current is to come from: the upstream side or the self-powered device. The issue comes when a self powered device pulls up an unpowered notebook.

Recommendations

⇒ Ensure that mobile system can tolerate the input current when it is off or in a zero-volt suspend.

5.1.3 Number of Ports on Notebook

Some of the chipsets support two ports. The present version of PC97 requires one USB port and recommends two. Support for only one USB port in the notebook system is recommended for the following reasons:

- Allows desktop equivalence.
- Limits the external load that USB devices can place on the notebooks to 2.5 watts.
- The second port on the chipset can be implemented on the docking station.
- When more than one port is required, use a self-powered hub, a device with a built-in hub, or a docking station to supply the additional ports.

5.1.4 Bus Segment Power Management

Explanation

Individual devices can not have their power state directly controlled using USB mechanisms. Rather USB manipulates the state of the bus segment upon which they reside. This means that a device's power state is most directly controlled by the state of the port it is attached to. For mobile this has two implications:

- The USB hub driver must play an active role in power management.
- USB does not define standard interface mechanisms for fine grain device power control.

Recommendations

Make sure that the hub driver used participates with the operating system's power management policy. Standardized individual USB device power control is being addressed in the USB Power Device Class working group in the form of an adjunct spec to the USB Core Specification.

5.1.5 Bus Powered Hubs - Power Supply Implications

Explanation

The bus powered hub must supply 5 volts at each of its USB connectors. A bus powered hub may have up to 4 connectors. When a USB device is first connected it presents an instantaneous load of up to 100 mA to the system. The USB specification requires that the device reach a stable state within 10ms which places certain stability requirements the power supply.

Recommendations

⇒ Ensure that the power supply meets the requirements, particularly when battery operated.

⇒ Provide the user with a BIOS option to disable USB when the system is battery operated.

⇒ Provide the user with an option which does not allow a bus-powered hub when the mobile system is battery operated.

⇒ On the Root Hub port, if an independent attachment notification mechanism is used (e.g. a switch in the USB connector), it may not be possible for the power supply to become stable within 10ms. In this case, the mobile system may allow the power supply to become stable over a longer period of time.

⇒ Any combination of the above.

5.1.6 Root Hub Port Power

Explanation

The USB specification says that each Root Hub port must be able to supply 500 mA. While this is a reasonable amount of power when AC operated, it can place too high a burden on the mobile system when battery operated. It is also possible that the system's power supply is simply unable to meet this requirement.

Recommendations

⇒ Make the USB system software stack aware of the system's power capabilities. This may include:

- Detection/notification of battery/AC operation (e.g. supply 100 mA when battery operated and 500 mA when AC operated)
- Low battery conditions
- Power supply limitations

Note: The USB system software stack may use the above information to effectively manage power as a bus resource.

6. Summary

Adding USB to the mobile platform provides desktop equivalence for this new external bus. USB's power consumption model is quite different than other add-in devices (i.e. PC Cards) used in mobile systems. There are several areas where an OEM can differentiate their platform's USB implementation.

The number and characteristics of ports on the mobile system can be defined to control the ultimate amount of power the mobile system must be able to supply to external devices. This may then affect system power supply requirements as seen in Table 4. One low-power port may be sufficient for most mobile systems.

Use of USB-independent mechanisms to notify the system when a USB device is attached to a Root Hub port can allow the mobile system to control power to the USB port. This can simplify the mobile system's power supply. When additional ports or high power device support is required, docking stations and self-powered devices and hubs can be used.

BIOS settings may be used to alter the platform's behavior and allow the user to select tradeoffs between performance and battery life. USB power can be turned off in order to maximize battery life at the expense of the availability of USB functionality. The BIOS can be used to set whether the Root Hub port is a high power or a low power port allowing the mobile system to control the amount of power it provides to external devices.

Internal USB devices should be avoided in mobile system designs. They can take more power than their existing counterparts.

USB software drivers can be supplied to manage the characteristics of the USB power consumption to allow for enhanced features while minimizing the overall power requirements. For example, replacing the hub driver with one that supports selective suspend will reduce the USB power use when resuming from a Remote Wakeup.

###