***Wake on USB* Feature (Mouse/Keyboard Wake Feature)**

**USB 2.0 Background Information**

The TPS2546 data lines interface with USB 2.0 devices. USB 2.0 defines three types of devices according to data rate. These devices and their characteristics relevant to TPS2546 *Wake on USB* operation are shown below.

* Low-speed USB devices
  + 1.5 MB/s
  + Wired mice and keyboards are examples
  + No devices that need battery charging
  + All signaling performed at 2.0V and 0.8V hi/lo logic levels
  + D- high to signal connect and when placed into suspend
  + D- high when not transmitting data packets
* Full-speed USB devices
  + 12 MB/s
  + Wireless mice and keyboards are examples
  + Legacy phones and music players are examples
  + Some legacy devices that need battery charging
  + All signaling performed at 2.0V and 0.8V hi/lo logic levels
  + D+ high to signal connect and when placed into suspend
  + D+ high when not transmitting data packets
* High-speed USB devices
  + 480 MB/s
  + Tablets, phones, music players are examples
  + Many devices that need battery charging
  + Connect and suspend signaling performed at 2.0V and 0.8V hi/lo logic levels
  + Data packet signaling performed a logic levels below 0.8V
  + D+ high to signal connect and when placed into suspend (same as a full-speed device)
  + D+ and D- low when not transmitting data packets

***Wake On USB***

*Wake on USB* is the ability of a wake configured USB device to wake a computer system from its S3 sleep state back to its S0 working state. *Wake on USB* requires the data lines to be connected to the system USB host before the system is placed into its S3 sleep state and remain continuously connected until they are used to wake the system. The TPS2546 data lines are connected in SDP/CDP modes and disconnected in DCP modes. Also, the data line connection will be broken whenever a TPS2546 mode change involving an OUT discharge occurs. The TPS2546 modes are controlled through the CTL pins. Theoretically, computer systems would implement *Wake on USB* with the TPS2546 by controlling the CTL lines appropriately to prevent a data line disconnect, when a USB device configured for *Wake on USB* is attached. In most computer systems, however, knowledge that a wake configured USB device is attached cannot be communicated to the circuits used to drive the CTL pins.

To overcome this computer limitation, the TPS2546 incorporates a *Wake on USB* feature. This feature identifies when a potential wake configured USB device is attached and overrides specific CTL pin changes that would break the data line connection. The TPS2546 uses device speed as a proxy to decide if a potential wake configured USB is attached. Low-speed and full-speed devices are classified as wake configured USB devices and are high-speed devices are not. The TPS2546 will identify when a low-speed or full-speed device is attached while in a SDP/CDP mode and override certain CTL pin changes to DCP-Auto to protect the data line connection. The specific CTL pin changes that the TPS2546 will override are shown below. The information is presented as CTL1, CTL2, CTL3. The ILIM\_SEL pin plays no role.

1) 111 (CDP/SDP2) to 011 (DCP-Auto)

2) 010 (SDP1) to 011 (DCP-Auto)

Note that the 110 (SDP1) to 011 (DCP-Auto) transition is not supported. This is done for practical reasons since the transition involves changes to two CTL pins. Depending on which CTL pin changes first, the device will see either a temporary 111 or 010 command. The 010 command is safe but the 111 command will cause an OUT discharge as the TPS2546 will instead proceed to the 111 state.

The TPS2546 *Wake on USB* feature limitations are due to the incomplete information it can discern about the attached device from its data speed. Devices wrongly identified will either not be able to perform a *Wake on USB* or charge in DCP-Auto. In practice, the number of devices adversely impacted will be small. Low-speed devices do not require battery charging, so there is no harm in always identifying as a wake configured USB device. Likewise, high-speed devices are usually never used to perform *Wake on USB* and almost always employ battery charging, so again there is little harm in never identifying as a wake configured USB device. Full-speed devices are more problematic since both wake configured USB devices and devices needing battery charging are found in this device type. Given that most full-speed devices requiring charging are older legacy devices (now replaced with high-speed devices) and wireless full-speed mice and keyboards that can be wake configured are very common, the TPS2546 identifies all full-speed devices as wake configured USB devices.

**USB Slow-Speed / Full-Speed Device Recognition**

The *Wake on USB* feature requires determining when a low-speed or full-speed device is connected or disconnected. USB hosts interact directly with attached devices to determine the device speed. The TPS2546 is unable to determine the device speed with this direct interaction. Instead, the TPS2546 passively monitors the D+ and D- data lines to recognize low-speed and full-speed devices.

For the below descriptions high means greater than 2V and low means lower than 0.8V. USB devices use differential signaling which means the data line are almost always at opposite polarities. The exception is when the host commands a reset to an attached device by driving both lines low.

Low-speed devices pull D- high and full-speed devices pull D+ high to signal connect, when placed into suspend, and when not transmitting data. High-speed devices pull D+ high to signal connect and when placed into suspend. After enumeration, the data lines of high-speed devices will always be lower than 0.8V. All USB 2.0 data lines will be below 0.8V when no device is attached. The TPS2546 follows the below statements to determine when a low-speed or full-speed device is connected or disconnected.

1. Connected if D- is high for at least 2ms

2. Connected if D+ is high for at least 60s

3. Disconnected if both D- and D+ are low for 64ms

The first statement identifies a low-speed device. The second statement identifies a full-speed device. The 60 seconds is a time integrated deglitch. The deglitch counts up if D+ is high and counts down if D- is low. The deglitch is needed to discriminate between full-speed and high-speed devices. The problem occurs when a high-speed device is attached during the computer transition from S0 to a sleep state. The computer will put the USB port into USB suspend before commanding the TPS2546 to a DCP-Auto mode. Since a high-speed device pulls D+ high when suspended, the danger is the device will be misclassified as a full-speed device and not allowed to proceed to sleep charging. The delay after the port is placed into suspend can be a few seconds for a transition to S3 and up to a few 10s of seconds for a transition to S4/S5. The deglitch addresses this concern. The trade-off is there is now a requirement that a full-speed device has to be connected for at least 60 seconds in SDP/CDP to be recognized as a potential wake configured USB device. There is no requirement that the full-speed device avoid data transmission for 60 seconds since a time averaged deglitch is used. In practice, the 60 second connection requirement will have little effect. Cases where a wake configured USB device needs to be identified in less than 60 seconds before the computer is placed in the S3 sleep state will be extremely rare.

Statement 3 identifies when a low-speed or full-speed device is disconnected. Once recognized, the TPS2546 does not track if the device was recognized as a low-speed or full-speed device. Thus, as long as either D+ or D- is held high, the device will still be deemed connected. Since USB signaling is differential, low-speed or full-speed devices will always have D+ or D- high except when held in reset by the host. The 64ms deglitch in statement 3 is designed to cover up these reset periods.

**No CTL Pin Timing Requirement After Wake Event and Transition from S3 to S0**

Unlike the TPS2543, there is no CTL pin timing requirement for the TPS2546 when the wake configured USB device wakes the system from S3 back to S0. The TPS2543 requires the CTL pins to transition from the DCP-Auto setting back to the SDP/CDP setting within 64ms of the attached USB device signaling a wake event (e.g. mouse clicked or keyboard key pressed). No such timing condition exists for the TPS2546.