1. PCI Standard Hot-plug Engineering Change Notice – SHPC Extensions for PCI-X 2.0

TITLE:	SHPC Extensions for PCI-X 2.0
DATE:	February 27, 2003
AFFECTED DOCUMENT(S):	PCI Standard Hot-Plug Controller and Subsystem Specification, Revision 1.0
SPONSOR:	Alan Goodrum, Hewlett-Packard, and Sridhar Muthrasanallur, Intel

1.1. Summary of the Functional Changes

Changes are to the PCI Standard Hot-Plug Controller and Subsystem Specification, Revision 1.0. This ECN extends the Standard Hot-plug Controller Specification to support the additional PCI-X speeds and modes allowed by the PCI-X 2.0 specification. Specifically, this ECN provides the required hardware and software extensions needed to support the new PCI-X 2.0 speeds of 266 and 533 and also the software extensions needed to control PCI-X 2.0 mode ECC and parity operation.

1.2. Benefits

The proposed changes provide the ability to support hot-plug functionality in PCI-X 2.0 systems in an industry standard manner.

1.3. Assessment of the Impact

The proposed changes to the Standard Hot-Plug Controller Specification will not affect existing systems and solutions based off of the SHPC 1.0 Specification. The proposed changes allow existing hardware to work fully with existing software. But the changes identified in this ECN are required for systems implementing a standard hot-plug solution for PCI-X 2.0. The proposed changes to the SHPC 1.0 specification would require new device and system hardware and new software.

1.4. Analysis of the Hardware Implications

PCI-X 2.0 systems can optionally implement a standard hot-plug solution based off of the SHPC 1.0 Specification along with the changes identified in this document. One additional pin (VIOSEL) is required from the standard hot-plug controller to support PCI-X 2.0 hot-plug.

1.5. Analysis of the Software Implications

A new SHPC device driver is needed to support standard hot-plug for PCI-X 2.0. A new SHPC device driver is required because of the programming interface changes required from the existing 1.0 specification. New hardware is not backward compatible with an SHPC 1.0 compliant device driver. A new SHPC device driver is required to work with new hardware. Old hardware can still continue to work with old software.

1.6. Additional Description and Rationale

PCI Standard Hot-Plug Controller and Subsystem Specification, Revision 1.0 only comprehends hot-plug in systems implementing conventional PCI and PCI-X 1.0 solutions. This specification needs to be extended to include the new speeds and modes introduced in the PCI-X 2.0 specification for an industry standard implementation of PCI-X 2.0 hot-plug. This ECN fills that need.

Details of changes

Change Instruction: Change reference to PCI-X 1.0a on pg 14 as follows:

~~"*PCI-X Addendum, Revision 1.0a* (PCI-X 1.0a), July 24, 2000, PCI Special Interest Group~~

"PCI-X Protocol Addendum to the PCI Local Bus Specification, Revision 2.0 (PCI-X PT 2.0), July 29, 2002, PCI Special Interest Group"

"PCI-X Electrical and Mechanical Addendum to the PCI Local Bus Specification Revision 2.0 (PCI-X EM 2.0), November 4, 2002, PCI Special Interest Group"

Change Instruction: Change all other references in the SHPC 1.0 document that refer to PCI-X 1.0a to PCI-X 2.0 as follows:

Page	Section	Change
14	1.3	Signal names defined in PCI 2.2, PCI-X PT 2.0 and PCI-X EM 2.0 ~~1.0a~~ and signals that are unique to hot-plug systems are indicated with **this bold** font.
15	1.3	This specification follows the precedent of PCI 2.2, PCI-X PT 2.0 and PCI-X -EM 2.0~~1.0a~~ by abbreviating clock frequency notation. For example, 33 1/3 MHz is written "33 MHz," 66 2/3 MHz is written "66 MHz," and 133 1/3 MHz is written "133 MHz."
20	2.2.1	The blink rate is permitted to be as low as 1 Hz in cases where the blink clock is derived from the PCI bus clock and the bus is operating at a speed that is lower than the maximum speed allowed by PCI 2.2 or PCI-X EM 2.0 ~~1.0a~~.
45	3	Although an SHPC must be integrated with a host or PCI-to-PCI bridge, this section only discusses device features that are unique to supporting hot-plug slots. Bridge features that are independent of the SHPC are controlled by PCI 2.2, PCI Bridge 1.1, and PCI-X PT 2.0 ~~1.0a~~.
45	3	The recommendations included in Section 3 have been selected to comply with the requirements of PCI 2.2, PCI Bridge 1.1, PCI-X PT 2.0, PCI-X EM 2.0 ~~1.0a~~, PCI HP 1.1, the Standard Usage Model in Section 2, and the programming interface in Section 4.
60	3.3/Table 3-3	**PCIXCAP** Input: A control logic input that indicates whether or not the add-in card is capable of supporting PCI-X mode and, if so, its maximum operating speed. See Section 4.5.12.1 and PCI-X EM 2.0 ~~1.0a~~ for additional requirements.
62	3.4	In addition, PCI 2.2 and PCI-X PT 2.0 ~~1.0a~~ require some bus

		signals to be driven prior to the deassertion of **RST#.** Hence, multiple SHPC output controls must change state in a prescribed sequence for the slot to be added or removed from the secondary bus such that:
62	3.4.1	Furthermore, when a slot is being brought out of the reset state (deassertion of **RST{n}#**), the SHPC drives appropriate initialization patterns as described in PCI 2.2 and PCI-X PT 2.0 ~~1.0a~~ to initialize the slot in a mode compatible with that of the secondary bus.
63	3.4.1	To initiate stabilization, the SHPC asserts **PHP_REQ#**. PCI-X PT 2.0 ~~1.0a~~ requires the arbiter to coordinate with the SHPC to keep the bus in an idle state while it drives the PCI-X initialization pattern before the rising edge of **RST{n}#**.
68	3.4.3	**Implementation Note: Clock Distribution** A PCI clock driver technique often used by Platform designs is to connect the outputs of multiple buffers together in order to minimize the clock skew seen at the PCI slots. This practice is discouraged for SHPC Platforms because glitches occurring on one hot-plug slot are seen by other slots. If the **CLK** signal at one slot violates the PCI 2.2 or PCI-X- EM 2.0~~1.0a~~ specification, the bus protocol state machines on enabled cards are potentially disrupted.
73	3.4.3	Note: PCI 2.2 and PCI-X EM 2.0 ~~1.0a~~ require some bus signals to be driven 10 clocks before RST# deassertion. The SHPC must insure BUSEN{n}# is asserted ahead of RST{n}# deassertion by 10 clocks plus additional clocks to adjust for bus connection latencies associated with the Slot Control Logic and bus switches (T_{bebc})
85	3.4.6	Recommended electrical parameters are listed in Table 3-12 and Table 3-13. These recommendations apply to the interfaces between the slot-specific circuit blocks shown in Figure 3-1andFigure 3-2. The serial interface between the SHPC and the Slot Control Logic is beyond the scope of this specification. Table 3-14 and Table 3-15 list recommended DC and AC specifications for the bus switches. See PCI-X EM 2.0 ~~1.0a~~ or PCI 2.2 for the electrical requirements of the following signals:
90	3.5.1	This section describes the various stimuli that initialize portions of a bridge integrated with an SHPC and specifies which portions are affected by which stimuli. When registers are initialized, they return to their default values specified in Sections 4.3, 4.5, PCI 2.2, PCI Bridge 1.1, and PCI-X PT 2.0 ~~1.0a~~. When state machines are initialized they return to a benign and stable state. In the case of a bridge that is integrated with an SHPC but is installed in an application in which the SHPC is disabled because it does not control any hot-plug slots (as described in Sections 4.3.1 and 4.3.2.4),

		the bridge is exempt from these requirements. Such a bridge is initialized exclusively as described in PCI 2.2, PCI Bridge 1.1, and PCI-X PT 2.0 ~~1.0a~~.
90	3.5.1.1	There are three stimuli that reset various portions of a bridge and its SHPC. 1. **Primary reset.** PCI 2.2, PCI Bridge 1.1, and PCI-X PT 2.0 ~~1.0a~~ require that **RST#** on the primary bus reset a PCI-to-PCI bridge's primary interface and Configuration Space registers, and that the signal propagate to the secondary bus **RST#**.

Change Instruction: Change section 2.2.1 page 20, as follows:

2.2.1 Indicators

The Standard Usage Model assumes that the Platform provides two indicators per slot (the Power Indicator (called "slot state indicator" in PCI HP 1.1) and the Attention Indicator). Each indicator is in one of three states: on, off, or blinking. Hot-plug system software has exclusive control of the indicator states by issuing commands to the SHPC. The SHPC controls blink frequency, duty cycle, and phase. Blinking indicators operate at a frequency of 1 to 2 Hz and 50% (+/- 5%) duty cycle. When more than one indicator is blinking, the SHPC is required to blink them synchronously and in-phase with each other for all indicators controlled by that SHPC. Blinking indicators are not required to be synchronous and in-phase if they are controlled by different SHPCs.

Implementation Note: Blinking Indicators

Ideally, the indicator blink rate should be 2 Hz if either of the following is true:

- The blink clock is derived from the PCI bus clock ~~and the PCI bus is operating~~ and the PCI bus clock is running at a speed of 33 MHz, 66 MHz, 100 MHz, or 133 MHz
- The blink clock is derived from a separate clock independent of the bus speed.
  The blink rate is permitted to be as low as 1 Hz in cases where the blink clock is derived from the PCI bus clock and the bus is operating at a speed that is lower than the maximum speed allowed by PCI 2.2 or PCI-X ~~1.0a.~~ EM 2.0

Change Instruction: Change Section 3.1.1 page 49, as follows:

3.1.1. Slot Control Logic

The Slot Control Logic is optionally designed to support either a serial or parallel SHPC interface. In either case, the Slot Control Logic is responsible for the following:

- Monitoring the power fault signal from the Power Controller and asynchronously asserting **RST{n}#** and disconnecting the slot from the secondary bus when a power fault occurs.
- Driving the slot-specific **M66EN{n}** signal as appropriate when the slot is enabled
- Analog-to-digital level conversion for the **PCIXCAP{n}** signal
  As illustrated in Figure 3-1, the SHPC uses a serial interface to communicate with external Slot Control Logic. Figure 3-2 illustrates an SHPC using a parallel slot interface

whereby the Slot Control Logic is connected to the SHPC in the bridge. Some high-frequency PCI-X implementations support only a single slot. For such implementations, the bus-switches are permitted to be integrated into the bridge.

If the SHPC supports a serial interface, the Slot Control Logic is also responsible for converting the serial interface signals to/from parallel form for use by the support electronics.

Change Instruction: Change Section 3.1.2.2 pg 51, as follows:

3.1.2.2 Power Fault Detection

The Power Controller monitors slot power for compliance with PCI 2.2. All SHPC Platforms must detect slot-specific over-current power fault conditions for each supply rail of each hot-plug slot. If the SHPC Platform supports **3.3Vaux**, over-current fault detection is also required on **3.3Vaux** of each hot-plug slot. All SHPC Platforms must detect slot-specific under-voltage conditions on the $V_{I/O}$ (when bus is operating in PCI-X Mode 2), 3.3V- and 5V-supply rails of each hot-plug slot.

The required 5V, 3.3V, $V_{I/O}$ (when bus is operating in PCI-X Mode 2), +12V, and –12V power rails are called "main power." The optional **3.3Vaux** supply is called "auxiliary power." The Power Controller detects and latches faults on main power independently from faults on auxiliary power. If a main power fault is detected, the Power Controller sets its internal main power fault latch and turns off main power to the slot (without affecting auxiliary power). The main power fault latch inside the Power Controller is cleared by the deassertion of the **PWREN{n}** signal. If the Power Controller supports **3.3Vaux** and an auxiliary power fault is detected, the Power Controller sets its internal auxiliary power fault latch, turns off auxiliary power to the slot (without affecting main power), and causes the **PME{n}#** and SMBus signals to be disconnected. (See Figure 3-18.) The auxiliary power fault latch inside the Power Controller is cleared when the slot's MRL is opened.

The Power Controller asserts the **PWRFLT{n}#** output if either or both internal power fault latches are set. See Section 3.1.1for the requirement for the Slot Control Logic to assert **RST#** and disconnect the slot when the **PWRFLT{n}#** signal asserts.

Change Instruction: Add a new Section after Section 3.1.2.3 pg 51, as follows:

3.1.2.4 Slot $V_{I/O}$ Select

The Power Controller receives a VIOSEL signal from the Slot Control Logic and selects the appropriate $V_{I/O}$ voltage level at the slot based on this signal (when PWREN signal to the slot is asserted). One value selects 3.3V for $V_{I/O}$ (conventional PCI mode and PCI-X Mode 1) and the other value selects 1.5V for $V_{I/O}$ (PCI-X Mode 2). All power controllers on a given bus segment are permitted to share the same VIOSEL signal from the Slot Control Logic.

This signal is only required if the SHPC supports PCI-X Mode 2 hot-plug slots and is not required if the SHPC supports only PCI-X Mode 1 hot-plug slots.

Note that since PCI-X 266 and PCI-X 533 cards never work in a 5V signaling environment, there is never a need for the VIOSEL signal to switch the $V_{I/O}$ at the slot between 5V and 1.5V.

Change Instruction: Change Section 3.2.4 pg 56, 57, as follows:

3.2.4 Bridge Capabilities, Capability Signals and REQ64#

If the secondary interface of the bridge integrated with an SHPC does not support PCI-X mode, the **PCIXCAP** input from the hot-plug slot is not required. In this case, the PCI-X Capability bits in the Slot Status field of the Logical Slot register are implemented as read-only and return a value of 0~~0b~~ (non-PCI-X) and the fields of the Slots Available registers must indicate no PCI-X slots are available. See Sections 4.5.12.1 and 4.5.2. If the secondary interface of the bridge integrated with an SHPC does not support 66 MHz conventional mode, the **M66EN** input from the hot-plug slot is not required. In this case, the 66 MHz Capable bit in the Slot Status field of the Logical Slot register is implemented as read-only and returns a value of 0 and the fields of the Slots Available registers must indicate no 66 MHz conventional-mode slots are available. See Sections 4.5.12.1 and 4.5.2.

If the bus segment controlled by the SHPC does not support 64-bit slots, the **REQ64#** pin of the slot (**REQ64{n}#**) must be pulled up to ~~slot power~~ $V_{I/O}$ when the bus is operating in parity mode. The **REQ64#** pins of each device must be bused when the bus is operating in ECC mode. (The **REQ64#** pin carries ECC information in ECC mode.)

If the bus segment controlled by the SHPC does not support PCI-X 266 or PCI-X 533, the VIOSEL signal to the power controller is not required. If the SHPC Programming Interface is 02h, but the SHPC does not support PCI-X 266 or PCI-X 533, the Slot Status field of the Logical Slot register must never indicate that the device in the slot is capable of PCI-X 266 or PCI-X 533 operation and the Slots Available registers must indicate no PCI-X 266 and PCI-X 533 slots are available.

Change Instruction: Change Section 3.3 pg 59, as follows:

3.3. Hot-Plug Controller Signals

The Slot Control Logic I/O signals described in Table 3-1 through Table 3-4 are unique (that is, slot-specific) for each supported slot, ~~that is, slot-specific,~~ except VIOSEL which is permitted to be shared by all power controllers. The I/O table heading indicates whether the signal is an input or output (or both) from the perspective of the Slot Control Logic. All SHPC I/O signals are required except the **CLKEN{n}#** signal. See Section 3.2.4 for cases for which **PCIXCAP** and **M66EN** are not required.

Table 3-1: Power Controller Interfaces

Signal Name	I/O	Definition
PWREN{n}	O	Power Enable Output: A Slot Control Logic output that controls the power state of the slot. If this signal is high, power is enabled to the slot.
VIOSEL	O	$V_{I/O}$ Select Output: A Slot Control Logic output that controls the $V_{I/O}$ voltage at the slot. If this signal is high, a $V_{I/O}$ voltage of 3.3V is selected and if this signal is low, a $V_{I/O}$ voltage of 1.5V is selected. This signal changes value when switching between conventional PCI/PCI-X Mode 1 and PCI-X Mode 2 (via the Set Bus Segment Speed/Mode command)
PWRFLT{n}#	I	Power Fault Input: A low-true input that indicates that the power controller for the slot has detected a power fault on one or more supply rails. See Sections 3.1.2.2 and 3.1.2.3 for additional requirements.

Change Instruction: Change Implementation Note in section 3.4.1 on pg 62, as follows:

3.4.1 Arbitration and Bus Stabilization

Bus arbitration and stabilization is recommended during the SHPC execution of commands that result in the assertion or deassertion of **CLKEN{n}#** or **BUSEN{n}#.** Bus arbitration and stabilization is required when the SHPC deasserts **RST{n}#**. The SHPC arbitrates for control of the bus in order to stabilize the secondary bus. For the purposes of this discussion, the signals **PHP_REQ#** and **PHP_GNT#** are used to interface to the secondary bus arbiter. In an actual implementation, these signals are permitted to be either package pins or internal connections.

When stabilizing the bus, it is recommended that the SHPC drive critical secondary-bus signals to a low-impedance, deasserted state. This is done to prevent glitches caused by a bus-switch event from occurring on the bus and potentially disrupting bus transactions to/from other enabled slots. Furthermore, when a slot is being brought out of the reset state (deassertion of **RST{n}#**), the SHPC drives appropriate initialization patterns as described in PCI 2.2 and PCI-X 1.0a to initialize the slot in a mode compatible with that of the secondary bus. The SHPC drives the following signals while stabilizing:

- **AD[31::00]**
- **C/BE[3::0]#**
- **PAR/ECC[0]**
- **FRAME#**
- **IRDY#**
- **DEVSEL#**
- **TRDY#**

- **STOP#**
- **PERR#** (when bus is operating in PCI-X Mode 2)
- **ECC[5::2]** (for buses that support ECC)
- **REQ64#/ECC[6]** (for 64-bit bus segments ~~only~~ and buses that support ECC)
- **ACK64#/ECC[1]** (for buses that support ECC)

Change Instruction: Change Section 3.4.3 on pg 67, 68, as follows:

Figure 3-5 through Figure 3-~~8~~ xx02 illustrate the bus timing requirements for bus arbitration and stabilization during Slot Enable command. It is recommended that the **PAR** signal is driven low and the other bus control signals (**FRAME#**, **IRDY#**, **TRDY#**, **DEVSEL#**, and **STOP#**), **AD[31::00]**, and **C/BE [3::0]#** are driven high by the SHPC during these phases, except as required for the PCI-X initialization pattern. Table 3-7 defines the recommended timing of bus control signals for phases 2 through 4 of the Slot Enable commands. Note that in PCI-X 266 and PCI-X 533, the arbiter is required to signal a turn-around alert on the bus before any device can issue a transaction on the bus. This is required to enable the newly added card to come out of interface low-power state and be ready to receive transactions.

Implementation Note: Reducing Bus Switch Charge Transfer
It is recommended that the SHPC drive the secondary bus signals high during a slot enable sequence that connects the bus. This is done to reduce the amount of charge transfer through the bus switches (when connection occurs) between the secondary bus and the pre-charged (high logic state) slot-side signals. When the bus is in PCI-X mode, a PCI-X initialization pattern must be driven on the **FRAME#**, **IRDY#**, **TRDY#**, **DEVSEL#**, **PERR#** and **STOP#** signals. Normally, the control signals are pulled up. However, driving the control signals high with a low impedance output buffer adds noise margin to the bus. The **AD[31::00]** and **C/BE [3::0]#** signals are also driven high since PCI 2.2, Section 3.4.3, requires the bus to be parked within eight clocks when the bus is idle. The **PAR** signal is driven low to maintain even parity on the bus. In PCI-X 66 with ECC, PCI-X 133 with ECC, PCI-X 266 and PCI-X 533, ECC[6::0] is driven to the ECC value corresponding to an address and command value of all 1s. If **FRAME#** signal integrity is corrupted and is seen asserted during the bus switch, a parity or ECC error will not be detected.

Change Instruction: Change Title of Figure 3-6 in Section 3.4.3 as follows:

Figure 3-6: PCI-X 66 and PCI-X 133 ~~Mode~~ SHPC Slot Enable Sequence Bus Stabilization Timing, Phase 2

Change Instruction: Add the following figure after Figure 3-6 in Section 3.4.3:

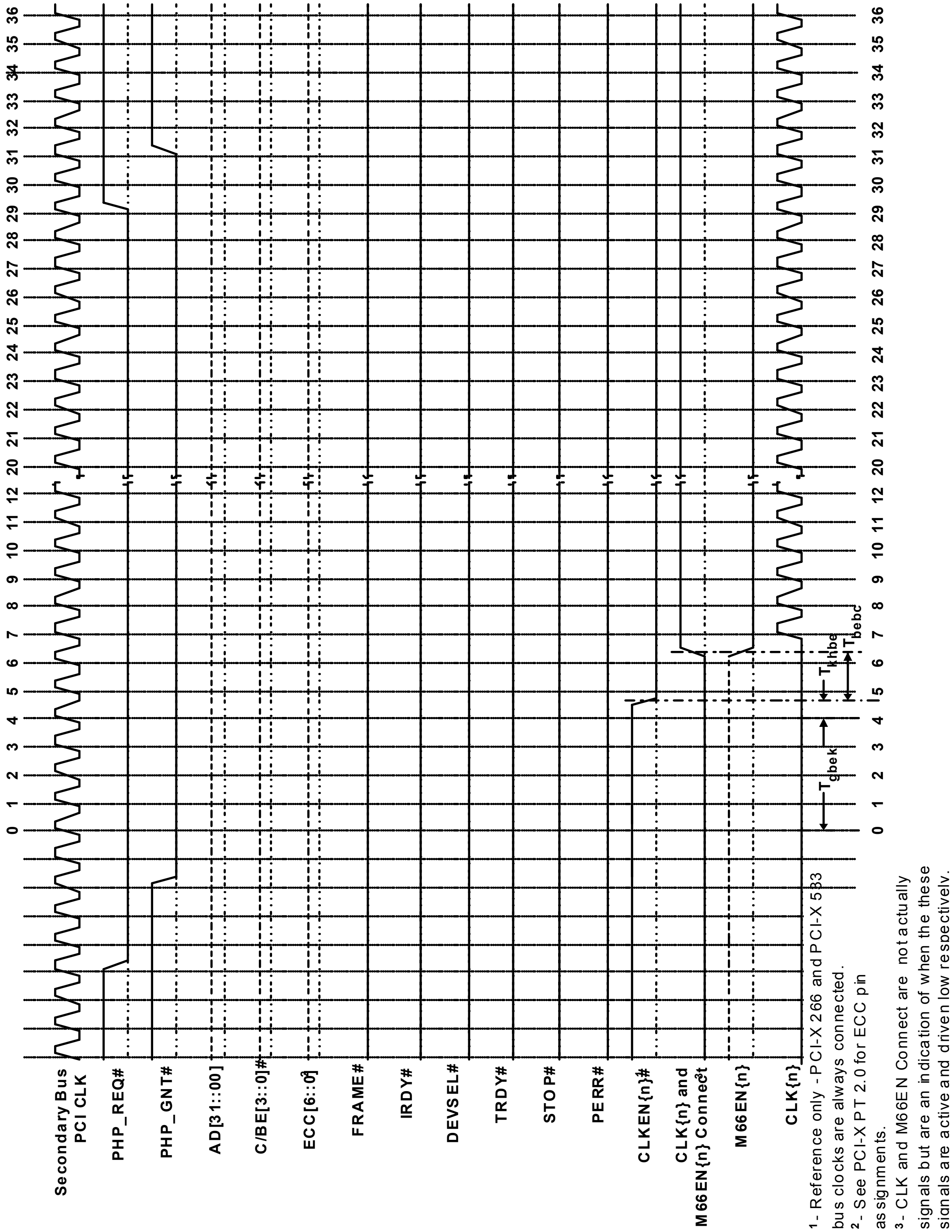


Figure 3-xx01: PCI-X 266 and PCI-X 533 SHPC Slot Enable Sequence Bus Stabilization Timing, Phase 2

Change Instruction: Change Title of Figure 3-8 in Section 3.4.3:

Figure 3-8: PCI-X 66 and PCI-X 133 ~~Mode~~ SHPC Slot Enable Sequence Bus Stabilization Timing, Phases 3 and 4

Change Instruction: Add the following figure after Figure 3-8 in Section 3.4.3:

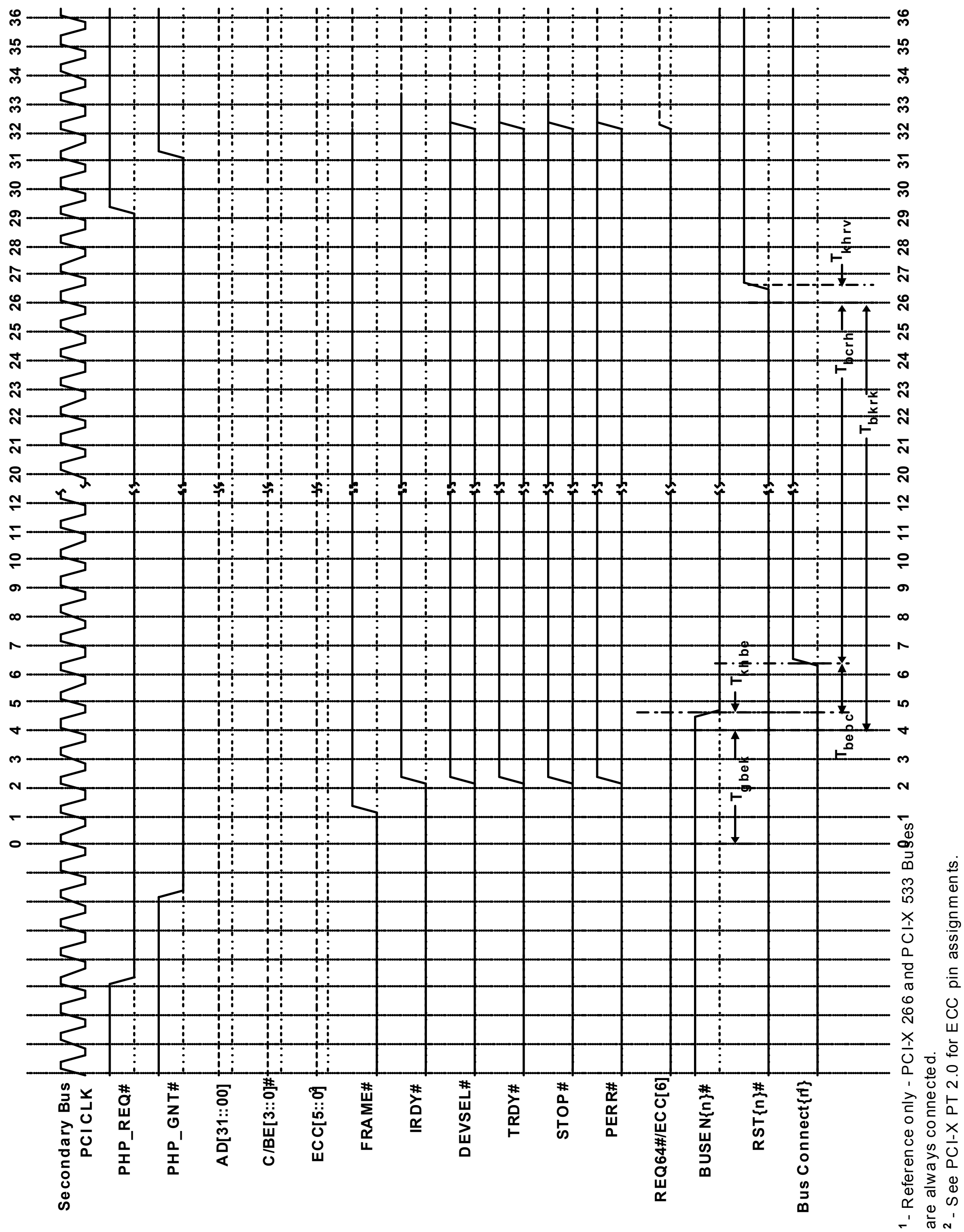


Figure 3-xx02: PCI-X 266 and PCI-X 533 SHPC Slot Enable Sequence Bus Stabilization Timing, Phases 3 and 4

Change Instruction: Change Title of Figure 3-11 in Section 3.4.4:

Figure 3-11: PCI-X 66 and PCI-X 133 ~~Mode~~ SHPC Slot Disable Sequence Bus Stabilization Timing, Phases 1 and 2

Change Instruction: Add the following figure after Figure 3-11 in Section 3.4.4 :

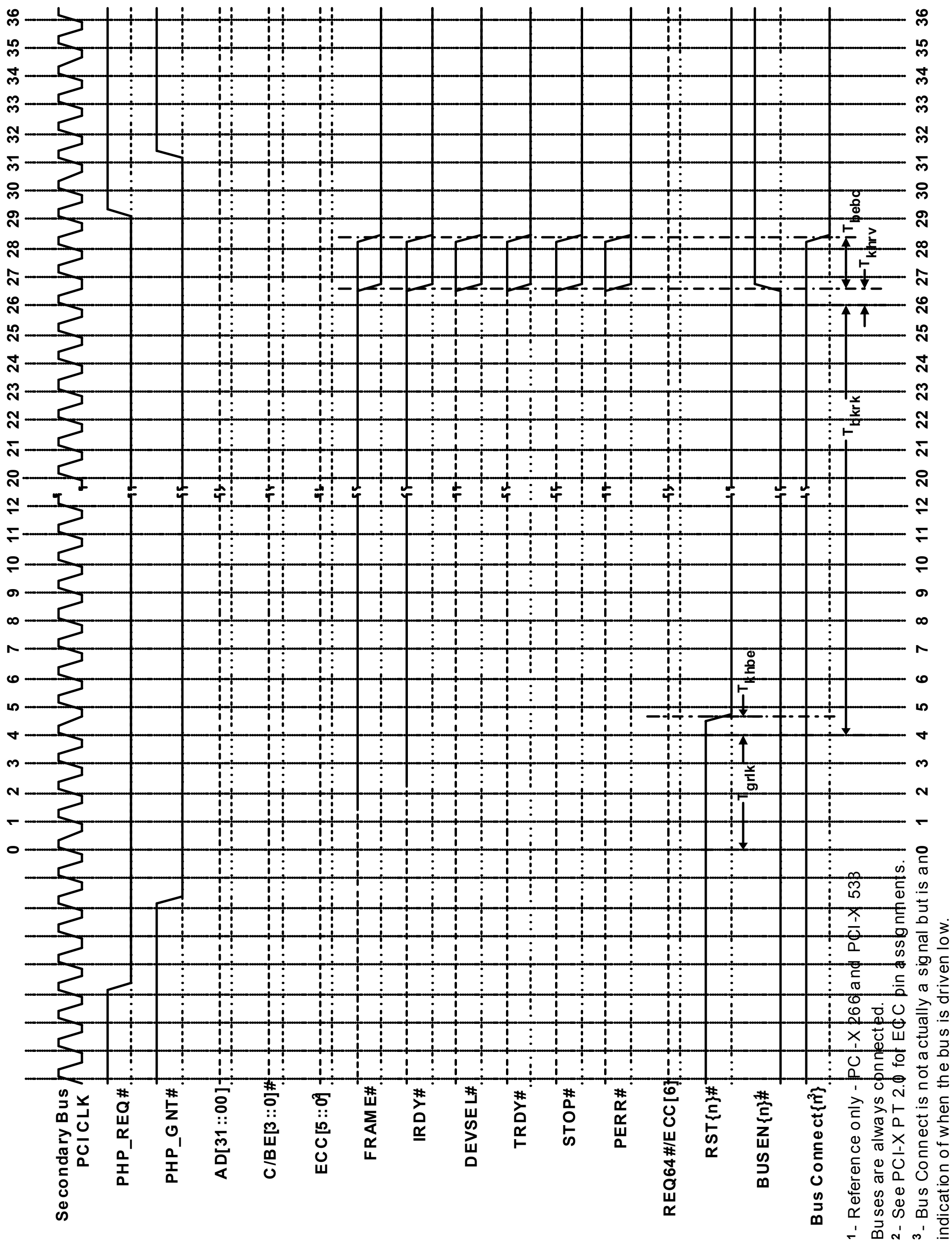


Figure 3-xx03: PCI-X 266 and PCI-X 533 SHPC Slot Disable Sequence Bus Stabilization Timing, Phases 1 and 2

Change Instruction: Change Title of Figure 3-13 in Section 3.4.4:

Figure 3-13: PCI-X 66 and PCI-X 133 ~~Mode~~ SHPC Slot Disable Sequence Bus Stabilization Timing, Phase 3

Change Instruction: Add the following figure after Figure 3-13 in Section 3.4.4:

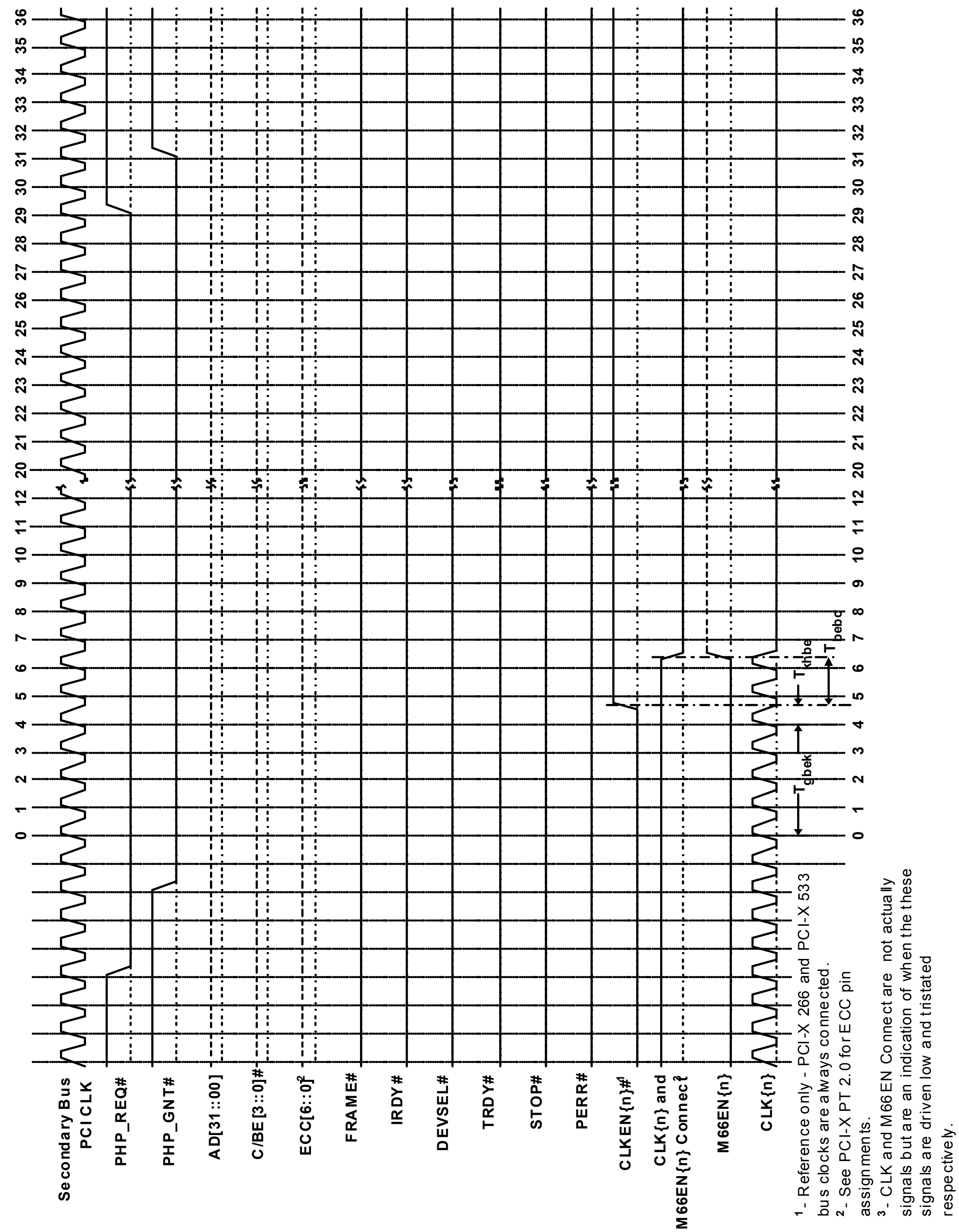


Figure 3-xx04: PCI-X 266 and PCI-X 533 SHPC Slot Disable Sequence Bus Stabilization Timing, Phase 3

Change Instruction: Change Table 3-7 in Section 3.4.3:

Table 3-7: SHPC Slot Enable and Disable Command Execution Sequence Bus Stabilization Timing

Timing Parameter	Description	Min	Max	Units	Reference Figures	Affected Components
T_{gbek}	Delay from bus idle-grant to **CLKEN{n}#** or **BUSEN{n}#** change	4	4	Secondary Bus Clocks	Figure 3-5 Figure 3-6 Figure 3-xx01 Figure 3-7 Figure 3-8 Figure 3-xx02 Figure 3-12 Figure 3-13 Figure 3-xx04	SHPC
T_{khbe}	Delay from secondary CLK to **CLKEN{n}#** or **BUSEN{n}#** change – output propagation delay	0	20	ns	Figure 3-5 Figure 3-6 Figure 3-xx01 Figure 3-7 Figure 3-8 Figure 3-xx02 Figure 3-10 Figure 3-11 Figure 3-xx03 Figure 3-12 Figure 3-13 Figure 3-xx04	Slot Control Logic

T_{bebc}	Delay from **BUSEN{n}#** or **CLKEN{n}#** changed to **M66EN{n}**, bus signal for slot n or **CLK{n}** connection or disconnection	7	60	ns	Figure 3-5 Figure 3-6 Figure 3-xx01 Figure 3-7 Figure 3-8 Figure 3-xx02 Figure 3-10 Figure 3-11 Figure 3-xx03 Figure 3-12 Figure 3-13 Figure 3-xx04	Bus Switches
T_{bcrh}	Delay from bus connection to **RST{n}#** deassertion (Note)	10	22	Secondary Bus Clocks	Figure 3-7 Figure 3-8 Figure 3-xx02	SHPC
T_{khrv}	Delay from **CLK** to **RST{n}#** change - output propagation delay	0	20	ns	Figure 3-7 Figure 3-8 Figure 3-xx02 Figure 3-10 Figure 3-11 Figure 3-xx03 Figure 3-15 Figure 3-16	Slot Control Logic
T_{bkrk}	Delay from **BUSEN{n}#** asserted to **RST{n}#** deasserted and delay from **RST{n}#** asserted to **BUSEN{n}#** deasserted	22	22	Secondary Bus Clocks	Figure 3-7 Figure 3-8 Figure 3-xx02 Figure 3-10 Figure 3-11 Figure 3-xx03	SHPC
T_{grlk}	Delay from bus idle-grant to **RST{n}#** asserted	4	4	Secondary Bus Clocks	Figure 3-10 Figure 3-11 Figure 3-xx03	SHPC

Change Instruction: Change Table 3-8 in Section 3.4.4:

Table 3-8: Standard Hot-Plug Controller Slot Disable Phases

Phase	Description	Signals Affected
1	During the first phase of a Slot Disable command, the **RST{n}#** signal asserts. Secondary bus arbitration is required for this phase. (See Figure 3-10, Figure 3-11 and Figure 3-~~11~~xx03 for bus stabilization requirements.)	**RST{n}#**
2	During the second phase of a Slot Disable command, the **BUSEN{n}#** signal is deasserted. Secondary bus arbitration is required for this phase. (See Figure 3-10, Figure 3-11 and Figure 3-~~11~~xx03 for bus stabilization requirements.)	**BUSEN{n}#**
3	During the third phase of a Slot Disable command, the **CLKEN{n}#** signal deasserts. Secondary bus arbitration is required for this phase. (See Figure 3-12, Figure 3-13 and Figure 3-~~13~~xx04 for bus stabilization requirements.)	**CLKEN{n}#**
4	During the fourth phase of a Slot Disable command, **PWREN{n}** deasserts. No bus arbitration is required for this phase. If indicator state changes have been requested with a Slot Disable command, the indicator state changes occur during this phase. (See Section 3.4.5, and Figure 3-14 for additional timing requirements.)	**PWREN{n}, ATNLED{n}#, PWRLED{n}#**

Change Instruction: Update other references to the newly added figures, as follows:

Page	Section	Change
66	3.4.3/Table 3-5	During the second phase of a Slot Enable command, the **CLKEN{n}#** signal asserts. Secondary PCI bus arbitration is required for this phase. During this phase, the slot-specific **CLK{n}** and **M66EN{n}** pin(s) of the slots are connected to the active bus segment. (See Figure 3-5, Figure 3-6 and Figure 3-~~6~~xx01.)
67	3.4.3	Figure 3-5 through Figure 3-~~8~~xx02 illustrate the bus timing requirements for bus arbitration and stabilization during Slot Enable command.
75	3.4.4	Figure 3-10 through Figure 3-~~13~~xx04 illustrate the bus timing requirements for bus arbitration and stabilization during Slot Disable commands. Table 3-7 defines the recommended timing of bus control signals for phases 1, 2, and 3. To simplify the implementation, the four phases for disabling a slot shown in Figure 3-9 are the inverse of the four phases for enabling a slot shown in Figure 3-4.
82	3.4.5	Arbitration is required when deasserting **RST{n}#** so that an appropriate initialization pattern can be driven as illustrated in Figure 3-15, Figure 3-16 and Figure 3-~~16~~xx05. Recommended timing is defined in Table 3-7 and Table 3-11.

Change Instruction: Change Title of Figure 3-16 in Section 3.4.5:

Figure 3-16: PCI-X 66 and PCI-X 133 ~~Mode~~ Deassertion of RST{n}# After a Bus Segment Reset or SHPC Set Frequency/Mode Command

Change Instruction: Add the following figure after Figure 3-16 in Section 3.4.5:

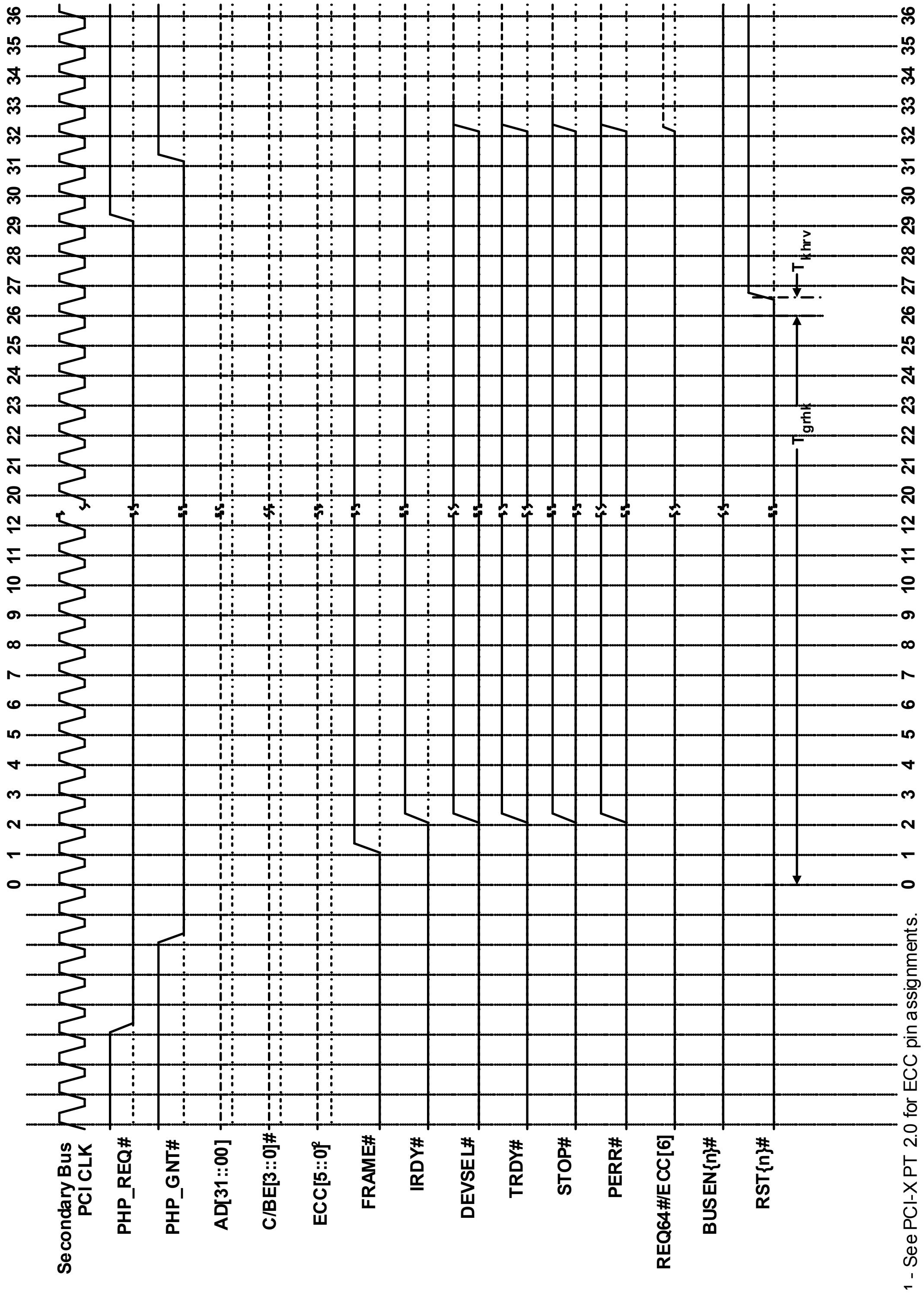


Figure 3-xx05: PCI-X 266, PCI-X 533 Mode Deassertion of RST{n}# After a Bus Segment Reset or SHPC Set Frequency/Mode Command

Change Instruction: Change Table 3-11 pg 81 in Section 3.4.5, as follows:

Table 3-11: Single-phase Timing Specifications

Timing Parameter	Description	Min	Max	Units
T_{pccc}	Delay from **PWREN** change to command completion. (See Figure 3-9 and Figure 3-14.)	25 (Note 1)	220 (Note 2)	ms
T_{grhk}	Delay from bus idle-grant to **RST{n}#** deassertion. (See Figure 3-15, Figure 3-16 and Figure 3-~~16~~ xx05.)	26	26	Secondary Bus Clocks

Change Instruction: Change Section 3.6.1 pg 95, as follows:

3.6.1. Power Fault Detection Requirements

Independent main power fault detection is required for all hot-plug slots. As a minimum, over-current fault detection is required for all main supply voltage rails, and under voltage fault detection is required for the $V_{I/O}$ (in PCI-X Mode 2 only), 3.3V and 5.0V main supply voltage supply rails. A power fault signal must be provided to the Slot Control Logic. Upon fault detection, the main power must be removed from the slot. The power fault signal must be latched (remain asserted) until the fault is explicitly cleared w hen the slot is turned off by softw are.

Change Instruction: Change Section 4.5.2 pg 112, as follows:

4.5.2 Slots Available Registers

The SHPC contains eleven~~five~~ Number of Slots Available fields organized into tw o Slots Available registers. Six of these fields correspond to PCI-X 266 and PCI-X 533 operation. Each Number of Slots Available field specifies the maximum number of hot-plug slots that are permitted to be enabled at the given speed and mode on the bus segment associated w ith this SHPC (see Figure 4-10 and Figure 4-xxxx). This gives the system a w ay to enforce bus-loading restrictions for different bus speed/mode combinations. For example, a value of 3 in the Number of Slots Available (100 MHz PCI-X) field limits the maximum number of slots that can be enabled on this bus segment to three, w hen the bus is operating in 100 MHz PCI-X mode. The same system w ould be permitted to set the Number of Slots Available (66 MHz PCI-X) to 4 indicating a maximum of four slots could be enabled at 66 MHz PCI-X mode. If a bus segment does not support any hot-plug slots at a given speed/mode, the corresponding Number of Slots Available field must be 0.

	~~31 29~~	~~28 24~~	~~23 21~~	~~20 16~~	~~15 13~~	~~12 8~~	~~7 5~~	~~4 0~~	~~Byte off.~~
~~Slots Available I~~	~~RsvdP~~	~~Number of Slots Available (133 MHz PCI-X)~~	~~RsvdP~~	~~Number of Slots Available (100 MHz PCI-X)~~	~~RsvdP~~	~~Number of Slots Available (66 MHz PCI-X)~~	~~RsvdP~~	~~Number of Slots Available (33 MHz Conv.)~~	~~04h~~
~~Slots Available II~~	~~RsvdP~~							~~Number of Slots Available (66 MHz Conv.)~~	~~08h~~

~~**Figure 4-10: Slots Available Registers (I and II)**~~

	31 29	28 24	23 21	20 16	15 13	12 8	7 5	4 0	Byte off.
Slots Available I	RsvdP	Number of Slots Available (133 MHz PCI-X)	RsvdP	Number of Slots Available (100 MHz PCI-X)	RsvdP	Number of Slots Available (66 MHz PCI-X)	RsvdP	Number of Slots Available (33 MHz Conv.)	04h

Figure 4-10: Slots Available Register I

	31 28	27 24	23 20	19 16	15 12	11 8	7 5	4 0	Byte off.
Slots Available II	Number of Slots Available (133 MHz PCI-X 533)	Number of Slots Available (100 MHz PCI-X 533)	Number of Slots Available (66 MHz PCI-X 533)	Number of Slots Available (133 MHz PCI-X 266)	Number of Slots Available (100 MHz PCI-X 266)	Number of Slots Available (66 MHz PCI-X 266)	RsvdP	Number of Slots Available (66 MHz Conv.)	08h

Figure 4-xxxx: Slots Available Register II (Note)

Note:
The register bits 31:8 are only defined as shown, if the SHPC Programming Interface register (PI) is 02h. They are reserved (of type RsvdP) if the SHPC Programming Interface register (PI) is 01h.

Table 4-7: Slots Available Registers I

Bit Location	Description	Register Attribute
4:0	**Number of Slots Available (33 MHz Conv.)** – Maximum number of hot-plug slots available to be enabled when the bus is running at 33 MHz conventional mode	HwInit
12:8	**Number of Slots Available (66 MHz PCI-X)** - Maximum number of hot-plug slots available to be enabled when the bus is running at 66 MHz PCI-X mode	HwInit
20:16	**Number of Slots Available (100 MHz PCI-X)** - Maximum number of hot-plug slots available to be enabled when the bus is running at 100 MHz PCI-X mode	HwInit
28:24	**Number of Slots Available (133 MHz PCI-X)** - Maximum number of hot-plug slots available to be enabled when the bus is running at 133 MHz PCI-X mode	HwInit

Table 4-8: Slots Available Registers II

Bit Location	Description	Register Attribute
4:0	Number of Slots Available (66 MHz Conv.) - Maximum number of hot-plug slots available to be enabled when the bus is running at 66 MHz conventional mode	HwInit
When PI=2		
11:8	Number of Slots Available (66 MHz PCI-X 266) - Maximum number of hot-plug slots available to be enabled when the bus is running at 66 MHz PCI-X 266.	HwInit
15:12	Number of Slots Available (100 MHz PCI-X 266) - Maximum number of hot-plug slots available to be enabled when the bus is running at 100 MHz PCI-X 266.	HwInit
19:16	Number of Slots Available (133 MHz PCI-X 266) - Maximum number of hot-plug slots available to be enabled when the bus is running at 133 MHz PCI-X 266.	HwInit
23:20	Number of Slots Available (66 MHz PCI-X 533) - Maximum number of hot-plug slots available to be enabled when the bus is running at 66 MHz PCI-X 533.	HwInit
27:24	Number of Slots Available (100 MHz PCI-X 533) - Maximum number of hot-plug slots available to be enabled when the bus is running at 100 MHz PCI-X 533.	HwInit

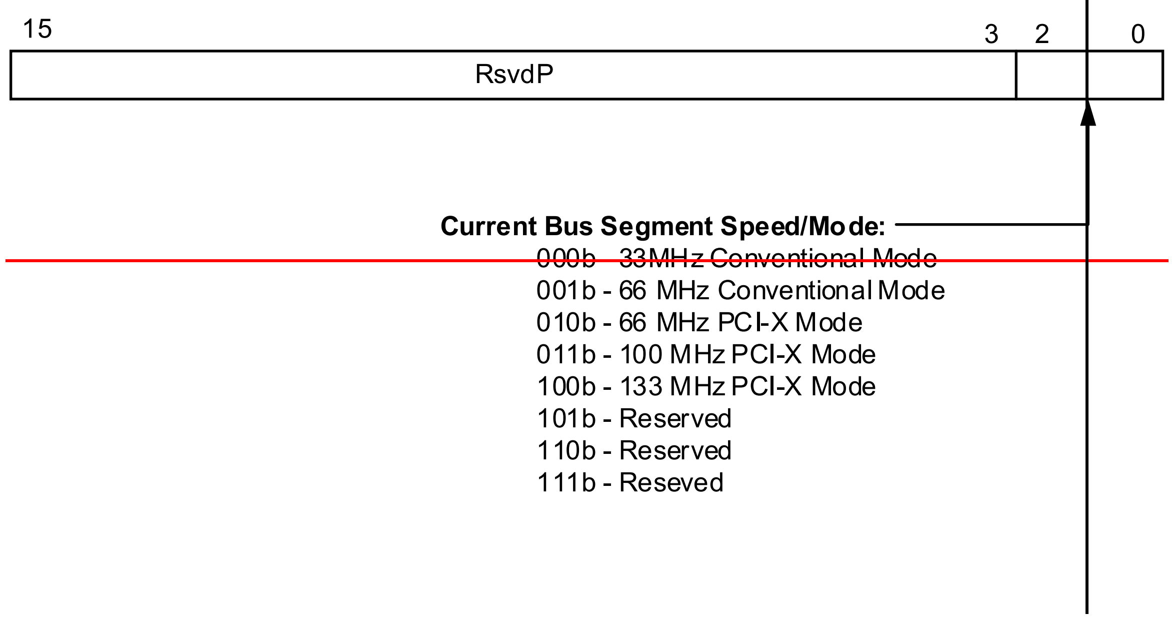
31:28	Number of Slots Available (133 MHz PCI-X 533) - Maximum number of hot-plug slots available to be enabled when the bus is running at 133 MHz PCI-X 533.	HwInit

Slots available at a particular bus speed/mode include only the first *n* slots controlled by the SHPC, where *n* is the value read from the Number of Slots Available field for that particular speed/mode. For example, assume a system supports four total slots and the slots start at device number 5. The Number of Slots Implemented field (see Section 4.5.3) would be set to 4 and the First Device Number field would be set to 5. Assume also that the bus segment supports two enabled slots at 100 MHz PCI-X mode ('Number of Slots Available (100 MHz PCI-X)' would be 2). The slots available at 100 MHZ PCI-X mode would include the slots with device numbers 5 and 6. Therefore, the SHPC must allow software to enable slots corresponding to device numbers 5 and 6, only. If software attempts to enable a slot corresponding to any other device number, the SHPC must cause the command to fail (see Section 4.6.1). Note that this places system design requirements on the placement of slots and their associated device numbers on a particular bus segment (see Section 4.1). The value in the Number of Slots Available field for a particular bus speed/mode must take into account the effect of non-hot-plug devices on the bus segment (both devices in non-hot-plug slots and embedded devices) as it relates to bus loading. See Section 3.2.7 for information on how systems support bus segments that contain both hot-plug and non-hot-plug devices.

Change Instruction: Change Section 4.5.4 pg 115, as follows:

4.5.4 Secondary Bus Configuration Register

The Secondary Bus Configuration register describes the configuration of the secondary bus segment that contains the hot-plug slots controlled by the SHPC. Bit definitions in this register are dependent on the value of the SHPC Programming Interface (PI) register at Byte Offset 13h.



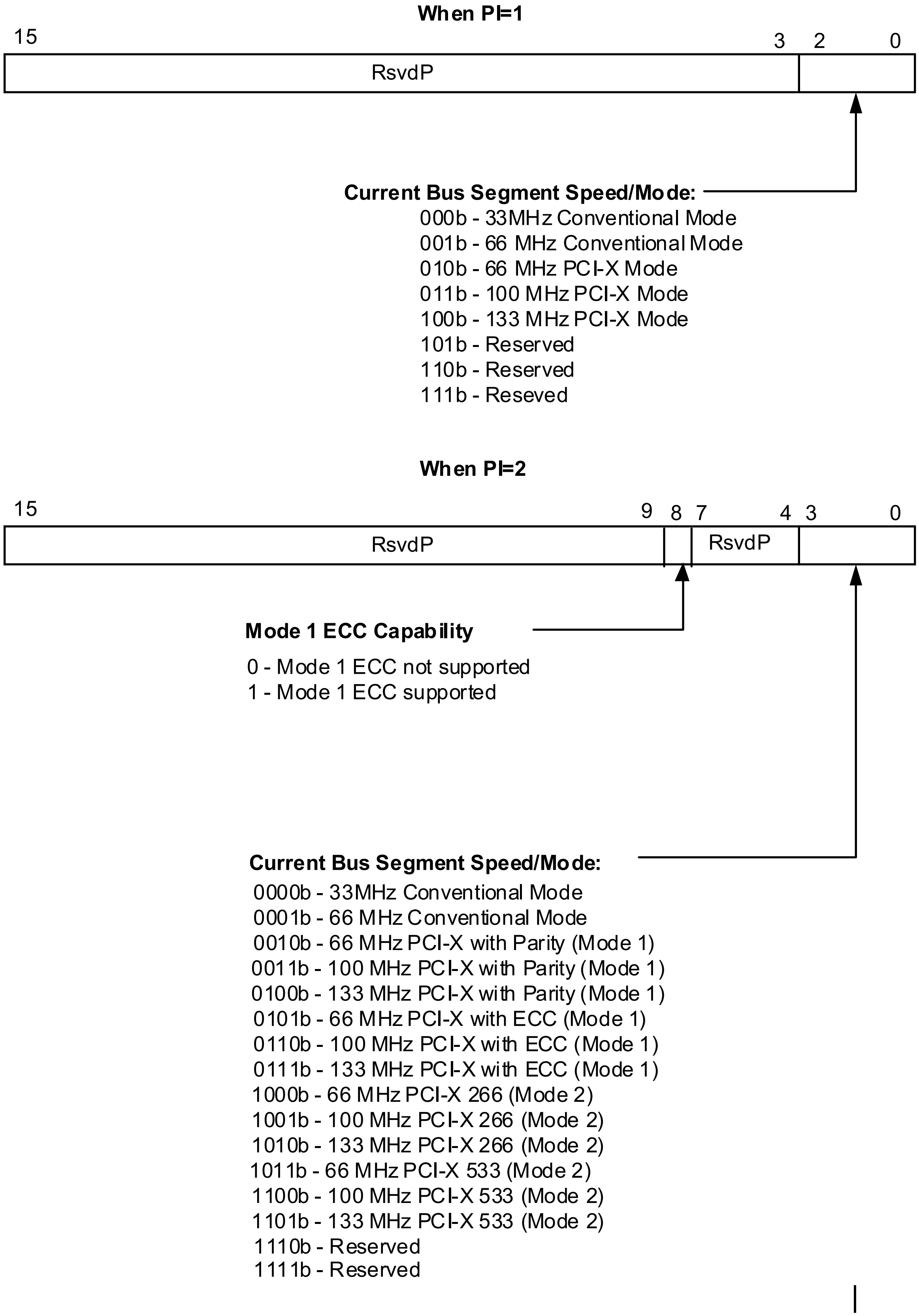


Figure 4-12: Secondary Bus Configuration Register

Table 4-10: Secondary Bus Configuration Register

Bit Location	Description	Register Attribute
When PI=1		
2:0	**Current Bus Segment Speed/Mode** – Indicates the current speed and mode at which the bus segment is operating.	RO
When PI=2		
3:0	**Current Bus Segment Speed/Mode** – Indicates the current speed and mode at which the bus segment is operating.	RO
8	**Mode 1 ECC Capability:** If this bit is 1, the bridge integrating the hot-plug controller supports ECC in PCI-X Mode 1. If this bit is 0, the bridge integrating the hot-plug controller supports only parity in PCI-X Mode 1. For PCI-X bridges that integrate an SHPC, this bit must be consistent with the Mode 1 ECC capability indicated in the PCI-X capability list item in the bridge's configuration register space	RO

Change Instruction: Change Section 4.5.6 pg 117 as follows:

4.5.6 SHPC Programming Interface (PI) Register

The SHPC Programming Interface identifies the format of the SHPC working registers.

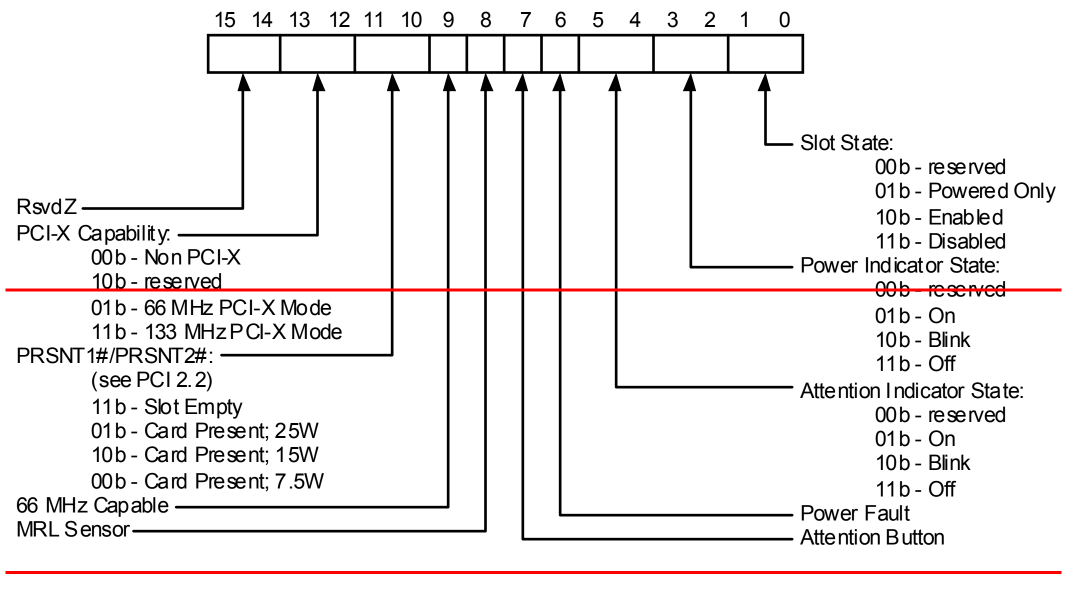
Table 4-12: SHPC Programming Interface Register

Bits Location	Description	Register Attribute
7:0	**SHPC Programming Interface** – Identifies the format of the SHPC Working Register set. ~~A~~ V~~v~~alues of 01h and 02h identify~~ies~~ the SHPC Working Register set formats defined in this specification. All other values are reserved. The two programming interfaces, 01h and 02h, are software incompatible. An SHPC designed for a programming interface of 02h can set its programming interfaces to 01h for software compatibility as long as it is prevented from being initialized to Mode 1 with ECC or Mode 2.	RO

Change Instruction: Change Section 4.5.12.1 pg 124, as follows:

4.5.12.1. Slot Status Field

The Slot Status field provides status information about a slot controlled by the SHPC.


15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Slot State:
00b - reserved
01b - Powered Only
10b - Enabled
11b - Disabled
RsvdZ
PCI-X Capability:
00b - Non PCI-X
10b - reserved
01b - 66 MHz PCI-X Mode
11b - 133 MHz PCI-X Mode
Power Indicator State:
00b - reserved
01b - On
10b - Blink
11b - Off
PRSNT1#/PRSNT2#:
(see PCI 2.2)
11b - Slot Empty
01b - Card Present; 25W
10b - Card Present; 15W
00b - Card Present; 7.5W
Attention Indicator State:
00b - reserved
01b - On
10b - Blink
11b - Off
66 MHz Capable
MRL Sensor
Power Fault
Attention Button

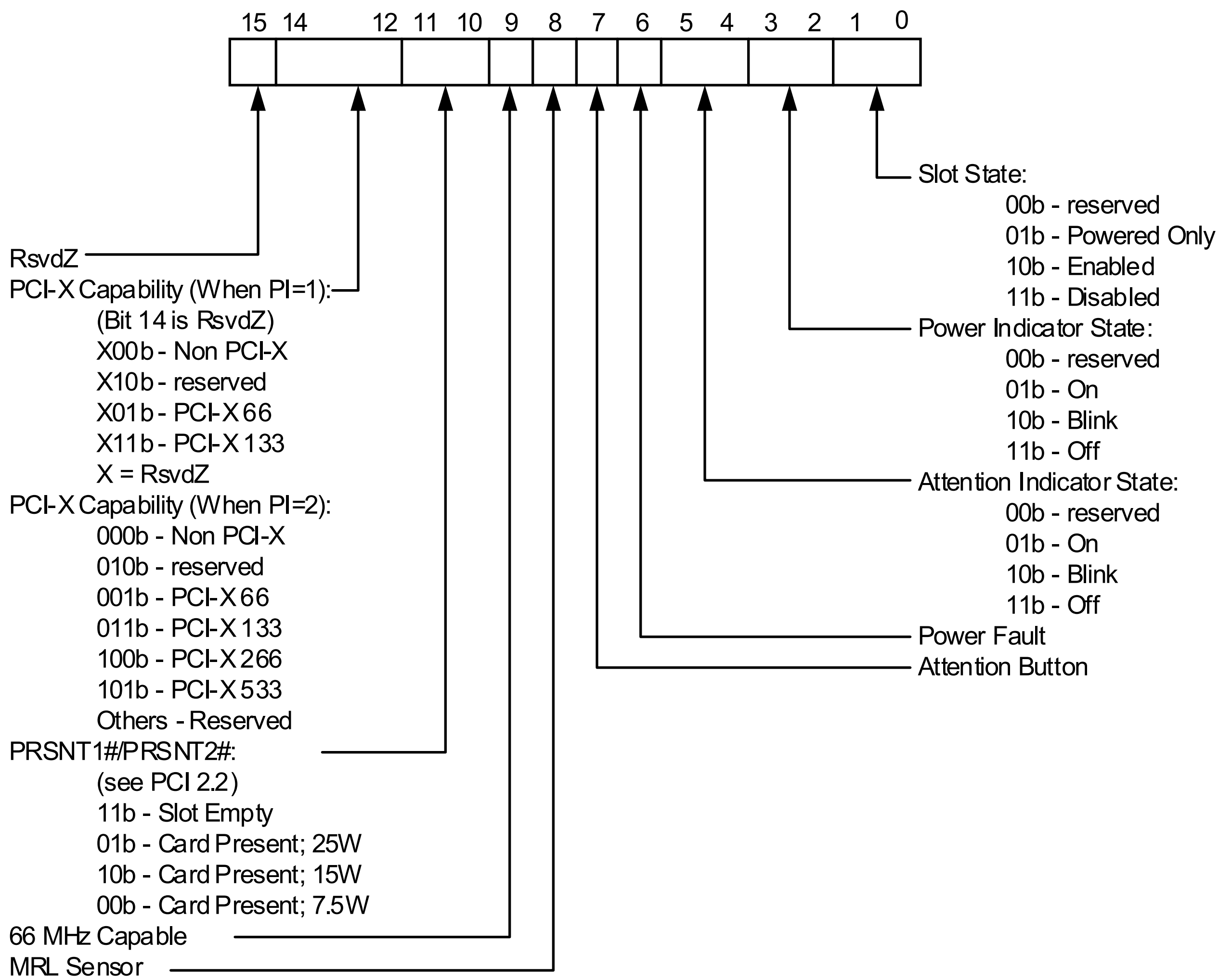


Figure 4-20: Slot Status Field

Table 4-18: Slot Status Field

Bit Location	Description	Register Attribute
1:0	**Slot State** – This field reports the current state of the slot (see Figure 4-20). This field is not affected by the detection of a slot power fault by the power controller circuitry (see Section 3.1.2.2). See Section 3.5.1.3 for information on the state of slots after coming out of reset.	RO
3:2	**Power Indicator State** – This field reports the current state of the Power Indicator associated with the slot (see Figure 4-20). See Section 3.5.1.3 for information on the state of Power Indicators after coming out of reset.	RO
5:4	**Attention Indicator State** – This field reports the current state of the Attention Indicator associated with the slot (see Figure 4-20). See Section 3.5.1.3 for information on the state of Attention Indicators after coming out of reset.	RO
6	**Power Fault** – This bit reports the current state of the power fault latch in the power controller circuitry for this slot. If this bit is 1, a power fault (either isolated or connected) has been detected by the power controller circuitry. See Section 3.1.2.3 for a description of power faults and how to clear the power fault latch in the power controller circuitry. See Section 4.7 for information about software handling of power faults.	RO
7	**Attention Button** – This bit reports the current state of the debounced Attention Button input signal for this slot (see Appendix A for a description of debounce). If this bit is 1, the Attention Button is being pressed. If this bit is 0, the Attention Button is not being pressed.	RO
8	**MRL Sensor** – This bit reports the current state of the MRL as reported by the debounced MRL Sensor input signal (see Appendix A for a description of debounce). If this bit is 1, the MRL Sensor is reporting that the MRL is open. If this bit is 0, the MRL Sensor is reporting that the MRL is closed.	RO

9	**66 MHz Capable** – This bit reports whether the add-in card is capable of running at 66 MHz conventional mode. This bit is latched as the slot is powered up or enabled, regardless of the current speed/mode of the bus (see Section 3.1.5). If this bit is 1, the card is capable of running at 66 MHz conventional mode. If this bit is 0, the card is only capable of 33 MHz conventional mode operation. This bit is valid only when the slot is occupied and powered or enabled.	RO
11:10	**PRSNT1#/PRSNT2#** – These bits report the current debounced state of the **PRSNT1#** and **PRSNT2#** pins on the slot (see Appendix A for a description of debounce). These bits are valid regardless of the state of the slot or speed/mode of the bus. See Figure 4-20 for possible values.	RO
When PI=1		
13:12	**PCI-X Capability** – These bits report the current PCI-X capability of the add-in card installed in the slot (see Figure 4-20). These bits are not valid if the slot is empty. If the slot is occupied, these bits are valid regardless of the state of the slot or speed/mode of the bus.	RO
When PI=2		
14:12	**PCI-X Capability** – These bits report the current PCI-X capability of the add-in card installed in the slot (see Figure 4-20). These bits are not valid if the slot is empty. If the slot is occupied, these bits are valid regardless of the state of the slot or speed/mode of the bus.	RO

Change Instruction: Change Table 4-21 in Section 4.6 pg 129 as follows:

Table 4-21: Controller Command Codes

Command	Command Code (hex)	Command Code Bits 7							0
Slot Operation	00h-3Fh	0	0	Attention Indicator		Power Indicator		Slot State	
Set Bus Segment Speed/Mode A	40h-47h	0	1	0	0	0	Bus Speed/Mode		
Power-Only All Slots	48h	0	1	0	0	1	0	0	0
Enable All Slots	49h	0	1	0	0	1	0	0	1
Reserved Command Codes	4Ah-4Fh	Reserved Command Codes							
Reserved Command Codes (When PI=1)	50-5Fh	Reserved Command Codes							
Set Bus Segment Speed/Mode B (When PI=2)		0	1	0	1	Bus Speed/Mode			
Reserved Command Codes	~~4A~~60h-BFh	Reserved Command Codes							
Vendor Specific Commands	C0h-FFh	1	1	Vendor-Specific Command					

Change Instruction: Change 4.6.1 in page 131 as follows:

Slot State Commands

The following commands are issued to change the state of a slot:

- Power Only – This command powers up the target slot but does not connect the clock or bus signals (see Section 3.4.5). Software uses this command to determine if a card is capable of running at 66 MHz conventional mode before enabling it (see Section 4.5.12.1).

  If the Target Slot field in the Controller Command register is greater than the Number of Slots Implemented field in the Slot Configuration register, the command must fail and the Invalid Command bit in the Controller Command Error Code field must be set.

  The slot must be disabled (or powered-only) when software issues this command. If the slot is currently enabled and this command is issued, the command must fail and the Invalid Command bit of the Controller Command Error Code field must be set.

  If the current bus mode/speed is PCI-X 266 or PCI-X 533 and the card being powered-only (and whose MRL is closed) is not capable of PCI-X 266 or PCI-X 533, then the command must fail and the Invalid Speed/Mode bit of the Controller Command Error Code field must be set.

If the MRL of the target slot is open, the command must fail and the MRL Open bit of the Controller Command Error Code field must be set.

- Enable – This command powers up the target slot, enables the clock, and connects the bus signals (see Section 3.4.3). At this point, the card is fully functional from a hardware standpoint. Software must still configure the card and load a driver for it.

  If the card is not capable of running at the speed or mode that the bus is currently running, the command must fail and the Invalid Speed/Mode bit in the Controller Command Error code field must be set. If the current mode/speed of the bus is PCI-X 266 or PCI-X 533 and the card being enabled (and whose MRL is closed) is not capable of PCI-X 266 or PCI-X 533, then the command must fail and the Invalid Speed/Mode bit in the Controller Command Error code field must be set without ever powering on the card.

Change Instruction: Change Section 4.6.2 in page 132 as follows:

4.6.2. Set Bus Segment Speed/Mode A and B

The Set Bus Segment Speed/Mode A and B commands changes the mode to conventional PCI, PCI-X Mode 1 or PCI-X Mode 2 and/or select the speed of the bus that contains the slots controlled by the SHPC. Also, in PCI-X Mode 1, separate command encodings are provided to initialize the bus segment in either parity or ECC modes. Usage of this command to select Mode 1 ECC operation is allowed only if the bridge and all devices on the bus support this mode. Software obtains this information for the bridge with the integrated SHPC by reading the Mode 1 ECC Capability bit in the Secondary Bus Configuration register of the SHPC working register set. Refer to the PCI-X PT 2.0 specification for details of how the PCI-X 2.0 devices report their Mode 1 ECC capability via the PCI-X capability register.

Issuing this the Set Bus Segment Speed/Mode A and B commands causes the bus segment to be reset and the requested bus speed and mode to take affect. Because add-in cards in the bus and its subordinate buses are reset by this command, software must quiesce all affected devices before issuing this command.

The lower three bits of the command code register for Set Bus Segment Speed/Mode A command and the lower four bits of the Command code register for Set Bus Segment Speed/Mode B command determine what new speed and mode the bus transitions to when the command completes (Table 4-23 and Table 4-XXXX). If the SHPC Programming Interface register is 02h, software is permitted to use either the Set Bus Segment Speed/Mode A Command or the Set Bus Segment Speed/Mode B Command to select conventional PCI mode or PCI-X Mode 1 with parity, as shown in Tables 4-23 and 4-XXXX.

Table 4-23: Format for Set Bus Segment Speed/Mode A Command

~~2~~	~~1~~	~~0~~	~~Requested Speed and Mode~~
~~0~~	~~0~~	~~0~~	~~33 MHz, Conventional Mode~~
~~0~~	~~0~~	~~1~~	~~66 MHz, Conventional Mode~~
~~0~~	~~1~~	~~0~~	~~66 MHz, PCI-X Mode~~
~~0~~	~~1~~	~~1~~	~~100 MHz, PCI-X Mode~~
~~1~~	~~0~~	~~0~~	~~133 MHz, PCI-X Mode~~
~~1~~	~~0~~	~~1~~	~~Reserved~~
~~1~~	~~1~~	~~0~~	~~Reserved~~
~~1~~	~~1~~	~~1~~	~~Reserved~~

2	**1**	**0**	**Requested Speed and Mode**
0	0	0	33MHz PCI, Conventional Mode
0	0	1	66MHz PCI, Conventional Mode
0	1	0	66MHz PCI-X with Parity (Mode 1)
0	1	1	100MHz PCI-X with Parity (Mode 1)
1	0	0	133MHz PCI-X with Parity (Mode 1)
1	0	1	Reserved
1	1	0	Reserved
1	1	1	Reserved

Table 4-XXXX[1]: Format for Set Bus Segment Speed/Mode B Command

3	**2**	**1**	**0**	**Requested Speed and Mode**
0	0	0	0	33 MHz PCI, Conventional Mode
0	0	0	1	66 MHz PCI, Conventional Mode
0	0	1	0	66 MHz PCI-X with Parity (Mode 1)
0	0	1	1	100 MHz PCI-X with Parity (Mode 1)
0	1	0	0	133 MHz PCI-X with Parity (Mode 1)
0	1	0	1	66 MHz PCI-X with ECC (Mode 1)
0	1	1	0	100 MHz PCI-X with ECC (Mode 1)
0	1	1	1	133 MHz PCI-X with ECC (Mode 1)
1	0	0	0	66 MHz PCI-X 266 (Mode 2)
1	0	0	1	100 MHz PCI-X 266 (Mode 2)
1	0	1	0	133 MHz PCI-X 266 (Mode 2)
1	0	1	1	66 MHz PCI-X 533 (Mode 2)
1	1	0	0	100 MHz PCI-X 533 (Mode 2)
1	1	0	1	133 MHz PCI-X 533 (Mode 2)
1	1	1	0	Reserved
1	1	1	1	Reserved

[1] This is only a place-holder for whatever table number will appear in the SHPC specification with this ECR incorporated

Software must make sure all add-in cards on the bus segment are capable of running at the requested speed before using this command. Card capabilities are determined using the 66 MHz Capable and the PCI-X Capable fields in the Slot Status field (see Section 4.5.12). Refer to the PCI-X PT 2.0 specification for details of how the PCI-X 2.0 devices report their Mode 1 ECC capability.

If any enabled add-in card on the bus segment is not capable of running at the requested speed or mode exclusive of parity/ECC capability, this command must fail, and the Invalid Speed/Mode bit in the Controller Command Error Code field must be set. (The SHPC does not validate the parity/ECC capability of the card.) Also, if the bridge integrating the standard hot-plug controller is incapable of running in PCI-X Mode 1 with ECC, then the corresponding Set Bus Segment Speed/Mode command that selects PCI-X Mode 1 ECC must fail, and the Invalid Command bit in the Controller Command Error Code field must be set.

If this command is issued to transition the bus segment to a speed/mode that would require a currently enabled add-in card to be disabled due to bus loading restrictions (as indicated by the appropriate Number of Slots Available field), the command must fail and the Invalid Speed/Mode bit must be set (see Section 4.5.2). For example, if all of the following are true:

- the bus segment is currently running at 100 MHz PCI-X mode
- the bus segment contains two enabled cards
- the Number of Slots Available (133 MHz PCI-X) field contains 1 then a command to change the bus to 133 MHz PCI-X mode must fail because the bus is not capable of supporting both enabled cards at 133 MHz PCI-X mode.

If the Set Bus Segment Speed/Mode command is issued for a segment and the Number of Slots Available register described in Section 4.5.2 indicates no slots are available at the requested speed/mode, the command must fail and the Invalid Command bit in the Controller Command Error Code field of the Controller Command Status register must be set. For example, if a bus segment is capable of 100 MHz PCI-X, the Number of Slots Available (100 MHz PCI-X) field would contain the number of slots that are permitted to be enabled on this bus segment. If this field is zero, the segment is not capable of running at 100 MHz PCI-X and a command to switch to it would fail as described above.

Software must disable all slots before changing a bus from conventional PCI or PCI-X Mode1 to PCI-X Mode 2, and vice versa. If a Set Bus Segment Speed/Mode B command that selects either PCI-X 266 or PCI-X 533 is issued, and at least one slot is enabled or powered-only, and the current bus segment mode is either conventional PCI or PCI-X Mode 1, the command must fail and the Invalid Command bit in the Controller Command Error Code field of the Controller Command Status register must be set. If a Set Bus Segment Speed/Mode A command or a Set Bus Segment Speed/Mode B command that selects either conventional PCI mode or PCI-X Mode 1 is issued and at least one slot is enabled or powered-only, and the current bus segment mode is PCI-X 266 or PCI-X 533, the command must fail and the Invalid Command bit in the Controller Command Error Code field of the Controller Command Status register must be set.

If this command fails, the speed and mode of the bus must not change and cards must not be reset or affected in any way.

Issuing this command with a value that matches the current speed/mode of the bus segment has the effect of resetting the bus. The speed and mode of the bus is not affected but add-in cards must be reset. Also, the controller follows the same rules for accepting or rejecting a Set Bus Segment Speed/Mode command (outlined earlier in this section), irrespective of whether there is at least one enabled slot or not.

All slot attributes must stay the same on completion of this command regardless of whether the command fails or succeeds.

The Target Slot bits in the Controller Command register must be ignored for Set Bus Segment Speed/Mode commands.

Change Instruction: Change Section 4.6.3 in page 133 as follows:

4.6.3. Power-Only All Slots

The Power-Only All Slots command changes the state of all slots controlled by the SHPC (that have their MRL closed) to "powered only" and changes the state of their Power Indicators to on. The number of slots controlled by the SHPC is determined by the Number of Slots Implemented field in the Slot Configuration register. Slots with their MRL open remain in the disabled state and their Power Indicator state is set to off (regardless of the indicators previous state). If all slots controlled by the SHPC have their MRLs open, the Slot States must remain disabled and the Power Indicator states must be set to off. (The command must complete successfully even if all the MRLs associated with controlled slots are open).

If one or more slots controlled by the SHPC are enabled when this command is issued, the command must fail. No slot states or LED states are changed, and the Invalid Command bit contained in the Controller Command Error Code field is set.

If the current mode/speed is PCI-X 266 or PCI-X 533 and any of the cards being powered-only whose MRL is closed is not capable of PCI-X 266 or PCI-X 533, then the command must fail and the Invalid Speed/Mode bit in the Controller Command Error Code field must be set.

Change Instruction: Change Section 4.6.4 in pages 133, 134 as follows:

4.6.4. Enable All Slots

The Enable All Slots command changes the state of all slots available at the current bus speed/mode (that have their MRL closed) to enabled and changes the state of their Power Indicator to on. See Section 4.5.2 for a description of all slots available at the current bus speed/mode. Slots with their MRL open remain in the disabled state and their Power Indicator state is set to off (regardless of the indicators previous state). If all slots available at the current bus speed/mode have their MRLs open, the slot states must remain disabled and the Power Indicator states must be set to off. (The command must complete successfully, even if all the MRLs associated with controlled slots are open.)

If one or more slots available at the current bus speed/mode are enabled when this command is issued, the command must fail. No slot states or LED states are changed, and the Invalid Command bit contained in the Controller Command Error Code field is set.

If one or more cards in an available slot at the current bus speed/mode has its MRL closed and does not support the present mode/speed of the bus, the command must fail and set the Invalid Speed/Mode bit in the Controller Command Error Code field.

This command can result in an "Invalid mode/speed" error in two cases:

- The user closes an MRL on a slot after software has checked mode/speed but before the Enable All Slots command is issued and that slot is not capable of running at the requested speed/mode.

- Software changes the bus mode/speed after the add-in card mode/speeds are checked, and an add-in card in a disabled slot does not support the new bus speed/mode.

If the current mode/speed of the bus is PCI-X 266 or PCI-X 533 and any of the cards being enabled whose MRL is closed is not capable of PCI-X 266 or PCI-X 533 operation. No power gets applied to any slot.

Change Instruction: Add the paragraphs below in 'Appendix B. Compliance Checklist' (Under 'SHPC Requirements' Subsection) on page 191 as follows:

After Item 52

If the current bus mode/speed is PCI-X 266 or PCI-X 533 when a Power-Only Command is issued and the card being powered-only (and whose MRL is closed) is not capable of PCI-X 266 or PCI-X 533, then the command must fail and the Invalid Speed/Mode bit of the Controller Command Error Code field must be set.

After Item 53

If the current Mode of the bus is PCI-X 266 or PCI-X 533 when an Enable Command is issued and the card being enabled (and whose MRL is closed) is not capable of PCI-X 266 or PCI-X 533, then the command must fail and the Invalid Speed/Mode bit in the Controller Command Error code field must be set, without ever powering on the card.

After Item 58

If a Set Bus Segment Speed/Mode B command that selects either PCI-X 266 or PCI-X 533 is issued and at least one slot is enabled or powered-only, and the current bus segment mode is either conventional PCI or PCI-X Mode 1, the command must fail and the Invalid Command bit in the Controller Command Error Code field of the Controller Command Status register must be set.

After Item 58

If a Set Bus Segment Speed/Mode A command or a Set Bus Segment Speed/Mode B command that selects either conventional PCI mode or PCI-X Mode 1 is issued and at least one slot is enabled or powered-only, and the current bus segment mode is PCI-X 266 or PCI-X 533, the command must fail and the Invalid Command bit in the Controller Command Error Code field of the Controller Command Status register must be set.

After Item 65

If the current mode/speed is PCI-X 266 or PCI-X 533 when a Power-Only All command is issued and any of the cards being powered-only whose MRL is closed, is not capable of PCI-X 266 or PCI-X 533, then the command must fail and the Invalid Speed/Mode bit in the Controller Command Error Code field must be set.

After Item 69

If the current mode/speed of the bus is PCI-X 266 or PCI-X 533 when an Enable All command is issued and any of the cards being enabled whose MRL is closed, is not capable of PCI-X 266 or PCI-X 533 operation. No power gets applied to any slot.

-- [END of ECR] --