PCI-SIG ENGINEERING CHANGE NOTIFICATION

TITLE:	REF CLK Delayed from CLKREQ# Assertion
DATE:	Initial request – June 30, 2010
AFFECTED DOCUMENT:	PCI Express Mini CEM Revision 1.2 and applicable ECNs published as of this notification.
SPONSOR:	Brad Saunders, Intel

Part I

1. Summary of the Functional Changes

This ECR requests making a change to the CLKREQ# asserted low to clock active timing when latency tolerance reporting is supported and enabled for the function. The change would be to allow this specified value to exceed 400ns up to a limit consistent with the latency value established by the Latency Tolerance Reporting (LTR) mechanism.

2. Benefits as a Result of the Changes

Allowing a delay in the reference clock enabling following CLKREQ# assertion will help to keep the power associated with the reference clock source lower longer when LTR is enabled. In effect, this will allow the system to take advantage of shutting more of the clock circuitry off for longer when it knows that additional delay is acceptable. Without the change, the reference clock would have had to be active within 400ns of CLKREQ# assertion and potentially much earlier than needed when latency tolerance values can be on the order of 10's of microseconds.

3. Assessment of the Impact

Add-in cards with functions that don't support LTR will not be affected. For those add-in cards with functions that do support LTR, the change is expected to be transparent given the system is by design to enable the reference clock with at least a minimum of the required L1 exit time for the function prior to the intended initiation of data transactions. The reason for this ECR is to formalize the behavior in order to help assure that add-in card functions aren't doing anything extraordinary related to the 400ns clock active timing, e.g. creating an internal error condition if the timing is not met – it is presumed that the 400ns value is only considered as an allocation in establishing the function's L1 exit value reported in configuration.

4. Analysis of the Hardware Implications

Presumed none as it is expected that existing endpoint functions should already be tolerant of a potential clock active delay. If this isn't the case, then such an endpoint function would have to be redesigned to tolerate the delay if the function is to be LTR compatible.

5. Analysis of the Software Implications

None. Existing LTR compatible software should already be compatible with this change although .

Part II – Detailed Description of the changes

Change the second paragraph of Section 3.2.4.2.2. as follows:

To exit L1, the device must assert CLKREQ# (low) to re-enable the reference clock. After the device asserts CLKREQ# (low) it must allow that the reference clock will continue to be in the parked clock state for a delay (T_{CRLon}) before transitioning to the active clock state. The time that it takes for the device to assert CLKREQ# and for the system to return the reference clock to the active clock state are serialized with respect to the remainder of L1 recovery. This time must be taken into account when the device is reporting its L1 exit latency.

When the PCI Express device supports and is enabled for Latency Tolerance Reporting (LTR), the device must allow that the reference clock transition to the active clock state may be additionally delayed by the system up to a maximum value consistent with requirements for the LTR mechanism. During this delay, the reference clock must remain parked. When exiting the parked state following the delay, the clock must be stable and valid within 400ns.

Change Table 3-3 as follows:

Table 3-3: CLKREQ# Clock Control Timings

Symbol	Parameter	Min	Max	Units
T_{CRHoff}	CLKREQ# de-asserted high to clock parked	0		ns
T_{CRLon}	CLKREQ# asserted low to clock active		400*	ns

* T_{CRLon} is allowed to exceed this value when LTR is supported and enabled for the device.