

PCI-SIG ENGINEERING CHANGE NOTICE

TITLE: Advanced Capabilities for Conventional PCI	
DATE:	13 April 2006, updated 27 July 2006
AFFECTED DOCUMENT:	PCI Specification Revision 3.0
SPONSOR:	Intel Corporation, Microsoft, HP

Part I

1 Summary of the Functional Changes

For conventional PCI devices integrated into a PCI Express Root Complex, this defines an optional capability structure that includes selected advanced features originally defined for PCI Express. This capability is intended to be extensible in the future. For the initial definition, the Transactions Pending (TP) and Function Level Reset (FLR) are included.

2 Benefits as a Result of the Changes

Selected new capabilities can be implemented in RC-integrated conventional PCI devices that cannot justify a conversion to PCI Express.

3 Assessment of the Impact

Moderate hardware implementation/validation impact for components implementing this optional capability.

4 Analysis of the Hardware Implications

Implementation and validation of a simple CSR structure. The implications of implementing TP and FLR are highly implementation specific and are likely to range from minimal impact to significant impact.

5 Analysis of the Software Implications

No impact to existing software. New software can optionally comprehend these new capabilities. Particularly for software written to support PCIe advanced capabilities, the impact will be minimal.

Part II

Detailed Description of the change

To the end of Chapter 6, following the MSI/MSI-X section, add section:

6.x. Advanced Features

For conventional PCI devices integrated into a PCI Express Root Complex, the Advanced Features (AF) capability provides mechanisms for using advanced features originally devleoped for PCI Express.

The Function Level Reset (FLR) mechanism enables software to quiesce and reset hardware with Function-level granularity.

FLR applies on a per Function basis. Only the targeted Function is affected by the FLR operation.

The Transactions Pending (TP) mechanism is used to indicate that the Function has issued one or more non-posted transactions (including Delayed Transactions) which have not been completed.

The FLR and TP mechanisms defined here are strictly for conventional PCI devices integrated into a PCI Express Root Complex where the implementation permits non-posted transactions for a given conventional PCI Function to complete even if the value of the Bus Master bit in its Command Register is 0. Implementations that do not meet this requirement must not implement the FLR and TP mechanisms.

6.x.1. Advanced Features Capability Structure

Figure 6.x shows the Advanced Features Capability Structure.

31	24	23 16	15 8	7 0
	AF Capabilities	LENGTH	NXT_PTR	CAP_ID
			AF Status	AF Control

Figure 6.x: Advanced Features Capability Structure

6.x.1.1. Capability ID for AF

<u>Bits</u>	<u>Field</u>	<u>Description</u>	
<u>7::0</u>	CAP_ID	The value of 13h in this field identifies the Function	
		as being AF capable. This field is read only.	

6.x.1.2. Next Pointer for AF

<u>Bits</u>	<u>Field</u>	Description
<u>7::0</u>	NXT_PTR	Pointer to the next item in the capabilities list. Must be NULL for the final item in the list. This field is read only.

6.x.1.3. AF Structure Length

<u>Bits</u>	<u>Field</u>	<u>Description</u>
<u>7::0</u>	<u>LENGTH</u>	AF Structure Length (Bytes). Shall return a value of 06h.

6.x.1.4. AF Capabilities

<u>Bits</u>	<u>Field</u>	<u>Description</u>
<u>7::2</u>	RESERVED	Reserved. Shall be implemented as read only returning a value of 00 0000b.
1	FLR_CAP	Set to 1b to indicate support for Function Level Reset (FLR). This field is read only.
<u>0</u>	TP_CAP	Set to 1b to indicate support for the Transactions Pending (TP) bit. TP must be supported if FLR is supported. This field is read only.

6.x.1.5. AF Control

<u>Bits</u>	<u>Field</u>	Description
<u>7::1</u>	RESERVED	Reserved. Shall be implemented as read only returning a value of 000 0000b.
<u>0</u>	INITIATE_FLR	A write of 1b initiates Function Level Reset (FLR). FLR requirements are defined in the PCI Express Base Specification. Registers and state information that do not apply to conventional PCI are exempt from the FLR requirements given there. The value read by software from this bit shall always be 0b.

6.x.1.6. AF Status

<u>Bits</u>	<u>Field</u>	<u>Description</u>
<u>7::1</u>	RESERVED	Reserved. Shall be implemented as read only returning a value of 000 0000b.
<u>0</u>	<u>TP</u>	Transactions Pending (TP): A value of 1b indicates that the Function has issued one or more non-posted transactions which have not been completed, including non-posted transactions that a target has terminated with Retry. A value 0b indicates that all non-posted transactions have been completed. This field is read only.

6.x.2. Advanced Features Operation

FLR modifies the Function state as follows:

Function registers and Function-specific state machines must be set to their initialization values as specified in this document, except for the following bits, which must not be modified: Fast Back-to-Back Enable, CacheLine Size, Latency Timer, Interrupt Line, PME En, PME Status.

Note that the controls that enable the Function to initiate bus transactions are cleared, including the Bus Master bit in the Command Register, the MSI Enable bit in the MSI Capability Structure, and the like, effectively causing the Function to become quiescent.

After an FLR has been initiated, the Function must complete the FLR within 100ms. If software initiates an FLR when the Transactions Pending bit is 1b, then software must not initialize the Function until allowing adequate time to achieve reasonable certainty that any outstanding transactions will have completed. The Transactions Pending bit must be clear upon completion of the FLR.

FLR modifies Function state not described by this specification (in addition to state that is described by this specification), and so the following criteria must be applied using Functionspecific knowledge to evaluate the Function's behavior in response to an FLR: The Function must not give the appearance of an initialized adapter with an active host on any external interfaces controlled by that Function. The steps needed to terminate activity on external interfaces are outside of the scope of this specification. For example, a network adapter must not respond to queries that would require adapter initialization by the host system or interaction with an active host system, but is permitted to perform actions that it is designed to perform without requiring host initialization or interaction. If the network adapter includes multiple Functions that operate on the same external network interface, this rule affects only those aspects associated with the particular Function reset by FLR. The Function must not retain within itself software readable state that potentially includes secret information associated with any preceding use of the Function. Main host memory assigned to the Function must not be modified by the Function. o For example, a Function with internal memory readable directly or indirectly by host software must clear or randomize that memory. The Function must return to a state such that normal configuration of the Function's PCI interface will cause it to be useable by drivers normally associated with the Function When an FLR is initiated, the targeted Function must behave as follows: The Function must complete normally the configuration write that initiated the FLR operation and then initiate the FLR. While an FLR is in progress:

targeting the Function will Master Abort).

The Function must not respond to any request on the bus (i.e. requests

The Transactions Pending (TP) bit indicates that the Function has issued one or more non-posted transactions which have not been completed. This field may be used by software to determine when a Function has become quiescent.



IMPLEMENTATION NOTE

Avoiding Issues with Pending Transactions

An FLR causes a Function to lose track of any pending (outstanding non-posted) transactions. Depending upon the specific implementation of the RC-integrated PCI Function, if software issues an FLR while there are pending transactions, there is a possibility for data corruption as described in the "Avoiding Data Corruption From Stale Completions" Implementation Note in the PCI Express Base specification.

To avoid potential issues with Root Complex implementations where Stale Completions are possible or a Discard Timer is present, it is recommended that software use an algorithm similar to the following:

- 1. Software that's performing the FLR synchronizes with other software that might potentially access the Function directly, and ensures that such accesses will not occur during this algorithm.
- 2. Software clears the entire Command register, disabling the Function from mastering any new transactions.
- 3. Software polls the Transactions Pending bit in the AF Status register either until it's clear or until it's been long enough to achieve reasonable certainty that any remaining outstanding Transactions will never complete. On many systems, the Transactions Pending bit will usually clear within a few milliseconds, so software might choose to poll during this initial period using a tight software loop. On rare cases when the Transactions Pending bit doesn't clear by this time, software will need to poll for a longer system-specific period (potentially seconds), so software might choose to conduct this polling using a timer-based interrupt polling mechanism.
- 4. Software initiates the FLR.
- 5. Software waits 100ms.
- 6. Software reconfigures the Function and enables it for normal operation.