



PCI-SIG ENGINEERING CHANGE NOTICE

TITLE:	PCI Option ROM CLP
DATE:	September 19, 2006
AFFECTED DOCUMENT:	PCI Firmware Specification, Revision 3.0
SPONSOR:	Darren Cepulis; Hewlett Packard Company

Part I

1. Summary of the Functional Changes

The definition of the CLP calling interface in the PCI Firmware 3.0 specification does not match calling conventions that have subsequently been defined by the DMTF. The PCI FW 3.0 specification defers to the DMTF CLP specification for coverage of differences, but is not specific in listing a particular DMTF specification (the Server Management Command Line Protocol Specification, Version 1.0) or set of supporting DMTF specifications (TBD).

This ECN rectifies the differences between the DMTF SM CLP Specification and its supporting documents with the current PCI Firmware 3.0 Specification. Also, it clarifies the supporting documents required for successfully implementing CLP in an X86 PCI FW 3.0 compatible option ROM.

2. Benefits as a Result of the Changes

This ECN enables developers by ensuring that the PCI Firmware specification will contain the most accurate and functional description of the PCI Option ROM CLP calling interface. Developers of systems and option cards that implement the DMTF SM CLP interface will benefit from a more concise interface and also from having a better description of the interface and how it is intended to be used.

3. Assessment of the Impact

This ECN proposes some changes to the register usage as part of the API of an option ROM's CLP entry point. There will be some slight change to system ROMs and option ROM's that already support this API.

Due to the new nature of the SM CLP interface and the lack of information available to implement a successful CLP implementation, changes to the Specification at this time should have minimal negative impact.

This ECN will not impact any implementation of the February 2006 ECN, since the return codes actually come from the same CIM_ERROR structure. Since this ECN really is just adding error values and the values are not overlapping with the values in the prior ECN, there is no impact on implementations above what the prior ECN would have had.

4. Analysis of the Hardware Implications

No impact to underlying hardware.

5. Analysis of the Software Implications

Functional compatibility of system firmware and PCI option ROM is increased due to the improved interface and description thereof.

1. Ambiguous register definitions in EBX have been dropped or redefined.
2. The use of the AX register for providing some targeting information has been added. AX is being defined to match the AX definition for the X86 PCI option ROM Init function entry point. It will contain the Bus/Dev/fcn of the source of the PCI option ROM. Although this may not completely obviate the need to parse the CLP command line for target information, it should simplify firmware and allow it to better determine the target of the CLP command being passed. This includes defining the contents of AH dependent on the value of AL. If AL is COMMAND PROCESSING FAILED, then AH should be the contents of Table 6. If AL is COMMAND EXECUTION FAILED, then AH should be the contents of Table 11. This would allow OEMs to also return an OEM specific code as well as the proper CLP code. Bit positions do not change.
3. Return values in AX have been added. These return values are clearly defined in DMTF standard tables in the SM CLP Specification. Additionally, an OEM specific return code in AX is allowed for, as needed.
4. ES:EDI pointer remains the same with added clarification.
5. DS:ESI pointer is added for the SM CLP Command Response string buffer, separate from the SM CLP Command string buffer pointed to by ES:EDI.
6. Deferral to DMTF standards body for future differences is dropped from footnote.
7. POST sequence clarified as it relates to PCI and SM CLP.
8. References to tables are by name and number instead of just number. This will allow the CLP spec to change without impact to the PCI FW Spec.

Part II

Detailed Description of the change

Note: Specification text is based on PCI Firmware Specification, Version 3.0, dated June 20, 2005 . All page numbers and section numbers mentioned below are based on that specification.

Change Section 5.1.2.25, page 91- 92 as follows:

5.2.1.25 DMTF Server Management Command Line Protocol (SM CLP) Support

The Option ROM may optionally provide an entry point that will support device configuration via the DMTF SM CLP standard.¹⁰ This interface will follow an API defined in the DMTF SM CLP Specification 1.0 Final Standard. This interface is accessed with a FAR CALL in Big Real mode where the system ROM will pass in a DMTF SM CLP compatible configuration message that will target the device or a child of that device. This interface may be called multiple times in order to completely configure a device during pre-boot. It is the responsibility of the System ROM to perform any discovery, enumeration, and subsequent translation of the SM CLP UFI's for any given Container in an implementation. The option ROM code should assume that the system firmware will call the entry point in Big Real Mode and with a minimum of 4 KB for the stack and at least 128KB of extended memory available via PMM.

Input parameters are described below:

~~ES:EDI = pointer to Command Line Protocol string buffer (NULL Terminated)~~

~~EBX = Command Status:~~

~~Bit 0: Privilege Bit:~~

~~0 = No Privileges~~

~~1 = Administrator Privileges~~

~~Bits 7-31: Reserved~~

~~Bits 15-8: Session ID~~

~~Bits 31-168: Process ID~~

Table 5-5: Input Arguments for SM CLP Entry Point

Argument Number	Register	Meaning
1	[AH]	Bus number
2	[AL]	Upper 5 bits are the Device number
3	[AL]	The lower 3 bits are the Function number
4	[ES:EDI]	Pointer to NULL-terminated SM CLP Command Line string buffer
5	[DS:ESI]	Pointer to SM CLP Command Response string buffer

AX contains the Bus/Device/Function of the source of the PCI option ROM, same as the PCI option ROM INIT and Configuration entry points described earlier in this specification. This provides an indicator of which PCI device is being targeted to the

SM CLP code. If a PCI option ROM's SM CLP entry point supports more than one device as an SM CLP target, the Command Line Protocol string buffer must also be examined for target information, according to the SM ME Addressing Specification. ES:EDI points to a Command Line string as defined in the SM CLP Specification (see section 3.1.9, "Input Data"). DS:ESI points to a buffer of at least 4K bytes in size. Strings are always NULL-terminated.

Note: The scope of the target in the SM CLP Command Line string buffer is limited to the Address Space of the PCI device and is not expected to be scoped to the overall containing Computer System. SM CLP implementations in PCI option ROMs are expected to adhere to the SM ME Addressing Specification with the exception that Collections, Logical Devices, Computer Systems, and other Managed Elements are allowed to be in the root of the Address Space.

The output parameters are described below:

~~ES:EDI = pointer to Command Line Protocol Return string buffer (NULL Terminated)~~

~~EBX = Return Status Flags~~

~~Bit 0: Success flag~~

~~0 = Command Accepted~~

~~1 = Command Failed~~

~~Bits 7-1: Reserved (always set to 0)~~

~~Bits 15-8: Return Status Code~~

~~0 = Command Completed Successfully~~

~~1 = Command Accepted, In Progress~~

~~2 = Invalid Target Specified~~

~~3 = Target Busy~~

~~4 = Insufficient Privilege~~

~~5 = Insufficient Resources~~

~~6 = Other Failure~~

~~7-255 = Reserved for future use~~

~~Bits 31-16: Process ID~~

Table 5-6: Output Arguments for SM CLP Entry Point

Argument Number	Register	Meaning
1	[AH]	If [AL] = 2 (COMMAND_PROCESSING_FAILED) the contents of [AH] are derived from the SM CLP Processing Error Value (see SM CLP Specification – Table 6: Processing Error Values and Tags). If [AL] = 3 (COMMAND_EXECUTION_FAILED) the contents of [AH] are derived from the SM CLP CIM Status Code Values (see SM CLP Specification – Table 9: CIM Status Code Values and Descriptions)
2	[AL]	SM CLP Command Status (see SM CLP Specification – Table 4: Command Status Values and Tags)
3	[EAX]	Bit 31: OEM Code Flag 0 = Execution Code is an SM CLP Probable Cause Value (see SM CLP Specification Table

		11: Probable Cause Values and Descriptions) 1 = Execution Code is an OEM Specific value
4	[EAX]	Bits 30-16: Execution Code
5	[ES:EDI]	Pointer to NULL-terminated SM CLP Command Line string buffer
6	[DS:ESI]	Pointer to NULL-terminated SM CLP Command Response string buffer

AL contains 0 for a successful command completion status. If AL is returned with a value of 2, a command processing error is signified and AH will contain a code signifying the cause of the processing error. If AL is returned with a value of 3, an execution error has occurred and the upper 16-bits of EAX will contain an execution error code. Bit 31 of EAX specifies whether the Execution Code is an SM CLP defined code or an OEM-specific code. The string buffer pointers (ES:EDI and DS:ESI) and the contents of the SM CLP Command Line string buffer are preserved. The SM CLP Command Response string buffer is filled in by the option ROM in the “keyword=value” format described in the SM CLP Specification (see section 3.1.10, “Output Data”), unless otherwise requested via the SM CLP –output option in the Command Line string buffer. Option ROM support for this default “keyword=value” output format is required, while support for other SM CLP output formats is optional (the SM CLP option ROM should return an OPTION NOT SUPPORTED error in AH if the SM CLP –output option requested by the caller is not supported by the option ROM).

All other x86 registers are preserved.

Note that not all PCI 3.0 system firmware will support calling the DMTF SM CLP entry point. The support for this function can be determined by examining the results of the “PCI BIOS Present” call described in Section 2.5.2. This is useful for diagnostics and validation in determining if the DMTF SM CLP interface will be used. In addition an Expansion ROM may change its configuration behavior if it determines that the PCI 3.0 system firmware will not call the DMTF SM CLP API. Similarly an Expansion ROM is not required to have a DMTF SM CLP entry point. If the entry point is not present, the “Pointer to the DMTF SM CLP entry point” field in the PCI Data Structure (see Section 5.2.1) should be null.

The normal power-up sequence concerning SM CLP is as follows:

- 1) An SM CLP configuration session is started via a remote console.
- 2) The system ROM facilitates the configuration of the PCI device(s) by passing SM CLP commands via the SM CLP Entry Point interface.
- 3) The Expansion ROM SM CLP code stores the configuration settings in a non-volatile location associated with the targeted PCI device so that the configuration information is available to the PCI device’s option ROM Init code on current and subsequent boots.
- 4) The system ROM initializes the PCI device(s) by calling the PCI Expansion ROM Init entry point.

Note: SM CLP configuration sessions that occur after the PCI Expansion ROM Init code has executed may require the system to be rebooted so that the new configuration changes can take effect.

Update Footnote at bottom of page 91 as follows:

10 The DMTF [SM](http://www.dmtf.org/standards/smash.apps/org/workgroup/svrmgmt/) CLP [S](http://www.dmtf.org/standards/smash.apps/org/workgroup/svrmgmt/)pecification is available from the DMTF group at <http://www.dmtf.org/standards/smash.apps/org/workgroup/svrmgmt/> in the Server Management Working Group. The PCI 3.0 Firmware Specification defers to the DMTF specification for any differences in description or implementation.