Document Number: DSP0261

Date: 2015-03-21

Version: 1.1.0

NC-SI over MCTP Binding Specification

Supersedes: 1.0.0

Document Type: Specification

Document Class: Normative

Document Status: Published

Document Language: en-US

CONTENTS

Figures

Tables

Foreword

The *NC-SI over MCTP Binding Specification* (DSP0261) was prepared by the Platform Management Components Intercommunications (PMCI Working Group) of the DMTF.

DMTF is a not-for-profit association of industry members dedicated to promoting enterprise and systems management and interoperability.

Acknowledgments

The DMTF acknowledges the following individuals for their contributions to this document:

Editor:

- Eliel Louzoun – Intel Corporation

Contributors:

- Alan Berenbaum – SMSC
- Philip Chidester – Dell
- Benzi Friedman – Intel Corporation
- Ed Klodnicki – IBM
- Patrick Kutch – Intel Corporation
- John Leung – Intel Corporation
- Hemal Shah – Broadcom Corporation
- Tom Slaight – Intel Corporation
- Bob Stevens – Dell
- Yuval Itkin – Mellanox

Introduction

The *NC-SI over MCTP Binding Specification* defines new MCTP messages used to convey NC-SI Control packets and Ethernet traffic over MCTP to allow NC-SI Pass-through traffic over MCTP. This specification is based on the DSP0222 1.0 specification and uses the same NC-SI Control packet definitions.

Document conventions

Typographical conventions

The following typographical conventions are used in this document:

- Document titles are marked in *italics*.
- Important terms that are used for the first time are marked in *italics*.
- Terms include a link to the term definition in the "Terms and definitions" clause, enabling easy navigation to the term definition.
- ABNF rules are in `monospaced font`.

ABNF usage conventions

Format definitions in this document are specified using ABNF (see RFC5234), with the following deviations:

- Literal strings are to be interpreted as case-sensitive Unicode characters, as opposed to the definition in RFC5234 that interprets literal strings as case-insensitive US-ASCII characters.

NC-SI over MCTP Binding Specification

1 Scope

The *NC-SI over MCTP Binding Specification* defines the bindings between NC-SI protocol elements and MCTP elements in order for NC-SI Control and Pass-through traffic to be transported using MCTP.

Portions of this specification rely on information and definitions from other specifications, which are identified in clause 2. Two of these references are particularly relevant:

- DMTF DSP0222, *Network Controller Sideband Interface (NC-SI) Specification*, provides the NC-SI base control that is to be bound over MCTP by this specification.
- DMTF DSP0236, *Management Component Transport Protocol (MCTP) Base Specification*, defines the MCTP transport on which the NC-SI Control and Pass-through packets are to be conveyed.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated or versioned references, only the edition cited (including any corrigenda or DMTF update versions) applies. For references without a date or version, the latest published edition of the referenced document (including any corrigenda or DMTF update versions) applies.

Unless otherwise specified, for DMTF documents this means any document version that has minor or update version numbers that are later than those for the referenced document. The major version numbers must match the major version number given for the referenced document.

DMTF DSP0004, *CIM Infrastructure Specification 2.6,*
http://www.dmtf.org/standards/published_documents/DSP0004_2.6.pdf

DMTF DSP0222, *Network Controller Sideband Interface (NC-SI) Specification 1.0*
http://www.dmtf.org/standards/published_documents/DSP0222_1.0.pdf

DMTF DSP0223, *Generic Operations 1.0*,
http://www.dmtf.org/standards/published_documents/DSP0223_1.0.pdf

DMTF DSP0236, *Management Component Transport Protocol (MCTP) Base Specification 1.2*
http://www.dmtf.org/standards/published_documents/DSP0236_1.2.pdf

DMTF DSP0237, *Management Component Transport Protocol (MCTP) SMBus/I2C Transport Binding Specification 1.0*
http://www.dmtf.org/standards/published_documents/DSP0237_1.0.pdf

DMTF DSP0238, *Management Component Transport Protocol (MCTP) PCIe VDM Transport Binding Specification 1.0*
http://www.dmtf.org/standards/published_documents/DSP0238_1.0.pdf

DMTF DSP0239, *Management Component Transport Protocol (MCTP) IDs and Codes 1.2*
http://www.dmtf.org/standards/published_documents/DSP0239_1.2.pdf

DMTF DSP1001, *Management Profile Specification Usage Guide 1.1,*
http://www.dmtf.org/standards/published_documents/DSP1001_1.1.pdf

ACPI, *Advanced Configuration and Power Interface Specification Revision 4.0a*, April 5, 2010
http://www.acpi.info/DOWNLOADS/ACPIspec40a.pdf

IETF, RFC4122, *A Universally Unique Identifier (UUID) URN Namespace*, July 2005
http://www.ietf.org/rfc/rfc4122.txt

IETF RFC5234, *ABNF: Augmented BNF for Syntax Specifications, January 2008*,
http://tools.ietf.org/html/rfc5234

ISO/IEC Directives, Part 2, *Rules for the structure and drafting of International Standards,*
http://isotc.iso.org/livelink/livelink.exe?func=ll&objId=4230456&objAction=browse&sort=subtype

3 Terms and definitions

In this document, some terms have a specific meaning beyond the normal English meaning. Those terms are defined in this clause.

The terms "shall" ("required"), "shall not," "should" ("recommended"), "should not" ("not recommended"), "may," "need not" ("not required"), "can" and "cannot" in this document are to be interpreted as described in ISO/IEC Directives, Part 2, Annex H. The terms in parenthesis are alternatives for the preceding term, for use in exceptional cases when the preceding term cannot be used for linguistic reasons. Note that ISO/IEC Directives, Part 2, Annex H specifies additional alternatives. Occurrences of such additional alternatives shall be interpreted in their normal English meaning.

The terms "clause," "subclause," "paragraph," and "annex" in this document are to be interpreted as described in ISO/IEC Directives, Part 2, Clause 5.

The terms "normative" and "informative" in this document are to be interpreted as described in ISO/IEC Directives, Part 2, Clause 3. In this document, clauses, subclauses, or annexes labeled "(informative)" do not contain normative content. Notes and examples are always informative elements.

Refer to DSP0236 for terms and definitions that are used across the MCTP specifications.

Refer to DSP0222 for terms and definitions that are used in the NC-SI specification.

The terms defined in DSP0004, DSP0223, and DSP1001 apply to this document. The following additional terms are used in this document.

3.1
Endpoint
An MCTP endpoint unless otherwise specified.

3.2
System Power States
S0 and Sx
S0 represents an active system
Sx represents system power states S1 – S5, which reflects various levels of inactivity of a system.

The definition of the power states is as defined in ACPI.

4 Symbols and abbreviated terms

The abbreviations defined in DSP0004, DSP0223, and DSP1001 apply to this document. The following additional abbreviations are used in this document.

4.1
ACPI
Advanced Configuration and Power Interface

4.2
IANA
Internet Assigned Numbers Authority

4.3
FCS
Frame Check Sequence

4.4
MCTP
Management Component Transport Protocol

4.5
MC
Management Controller

4.6
NC
Network Controller

4.7
NC-SI
Network Controller Sideband Interface

4.8
RID
PCIe Requester ID (Bus/Device/Function).

5 Conventions

5.1 Reserved and unassigned values

Unless otherwise specified, any reserved, unspecified, or unassigned values in enumerations or other numeric ranges are reserved for future definition by the DMTF.

Unless otherwise specified, numeric or bit fields that are designated as reserved shall be written as 0 (zero) and ignored when read.

5.2 Byte ordering

Unless otherwise specified, byte ordering of multibyte numeric fields or bit fields is "Big Endian" (that is, the lower byte offset holds the most significant byte, and higher offsets hold lesser significant bytes).

6 NC-SI over MCTP overview

NC-SI over MCTP is based on DSP0222 (NC-SI). The *NC-SI over MCTP Binding Specification* replaces the RBT Protocol with a definition of NC-SI communications using MCTP. The MCTP Transport Bindings are defined in other companion specifications such as *MCTP SMBus Binding Specification* (DSP0237) and *MCTP PCIe Binding Specification* (DSP0238). Only the NC-SI command processing is inherited from DSP0222. Thus only parts of the NC-SI specification not related to the physical transport protocol may be relevant to this specification.

6.1 NC-SI over RBT

A Network Controller Sideband Interface (NC-SI) is a combination of logical and physical paths that interconnect the Management Controller and Network Controller(s) for the purpose of transferring management communication traffic among them. NC-SI includes commands and associated responses, which the Management Controller uses to control the status and operation of the Network Controller(s). NC-SI also includes a mechanism for transporting management traffic and asynchronous notifications.

Figure 1 depicts the NC-SI Traffic Flow Diagram as currently defined by NC-SI. As indicated, the interface is based on RBT. The figure depicts a single management controller and a single Ethernet device, which contains a single port. NC-SI comprehends multiple Network Controller devices (or “packages”) and ports (or “channels”).

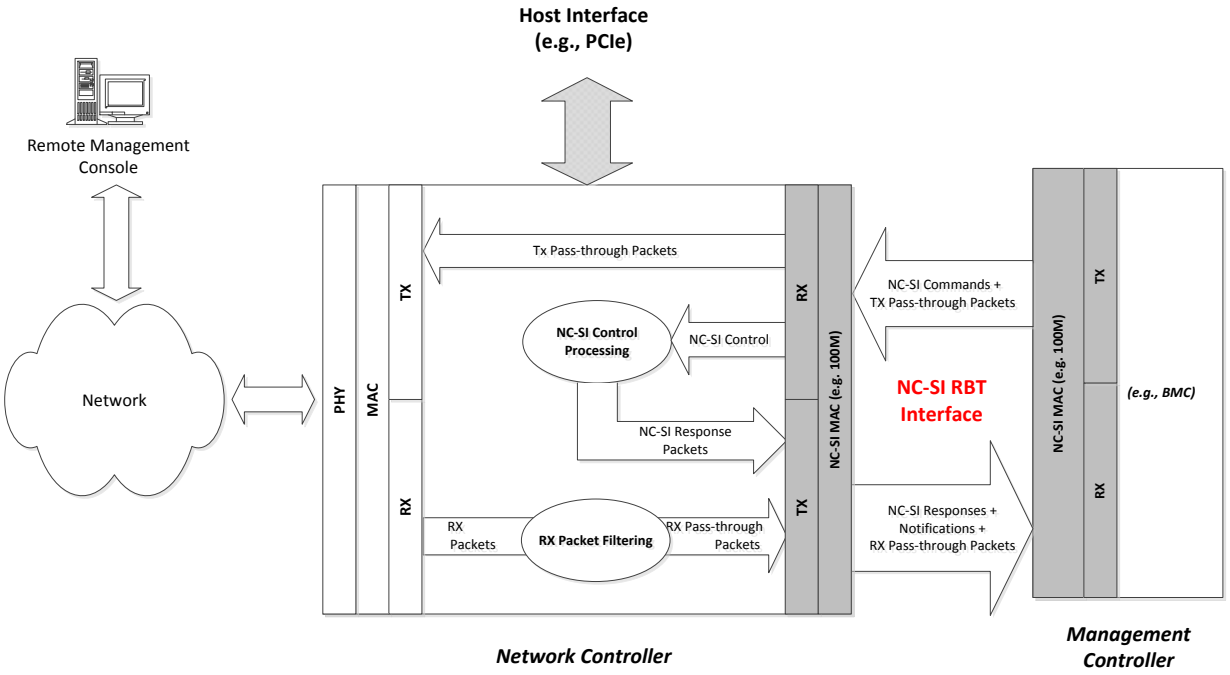

Figure 1 – NC-SI over RBT traffic flow diagram

The DSP0222 specification can be divided in three parts. The first two parts are defined as RBT:

- A physical layer based on enhancements to the RMII specification.
- A transport layer based on Ethernet packets. This layer allows differentiation of control frames based on a specific Ethertype (0x88F8).
- A control protocol defining a set of commands allowing an MC to configure and monitor Network Controllers and their Pass-through channels for MC to network communication. The command set functionality can be extended using OEM commands.

6.2 NC-SI over MCTP

NC-SI over MCTP replaces the transport layer defined in NC-SI with MCTP. The physical layer used is one of the transport bindings on which MCTP can be bound (for example, PCIe or SMBus).

Figure 2 shows a possible architecture that provides equivalent functionality to [NC-SI] over MCTP. The NC-SI MAC block in each device is replaced by an MCTP block and a Medium-specific block. The MCTP block handles MCTP messages. The Medium-specific blocks consist of whatever layers are involved in mapping MCTP to an underlying medium such as SMBus, PCIe, or USB. Because the layering for each medium may be unique in its constitution and terminology, a generic single block is depicted.

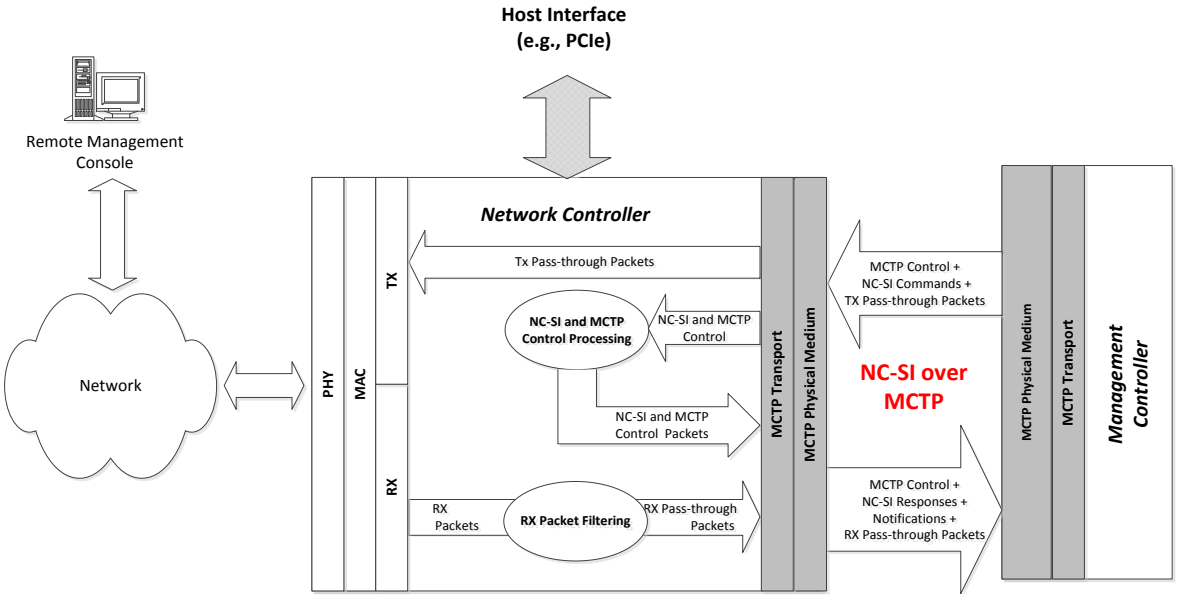

Figure 2 – NC-SI over MCTP traffic flow diagram

The differentiation between NC-SI Control and Pass-through packets is achieved by using two different MCTP message types as defined in DSP0239 and listed in Table 1.

306 **Table 1 – MCTP Message types for NC-SI over MCTP**

Message Type	Message Type Code	Description
NC-SI Control	0x02	Messages used to encapsulate NC-SI Control traffic (commands, responses, and AEN) over MCTP
Ethernet	0x03	Messages used to encapsulate Ethernet traffic (for example, NC-SI Pass-through) over MCTP

307

308 Both NC-SI Control and Ethernet types of MCTP messages can be conveyed over multiple MCTP
309 packets.

310 The encapsulation of NC-SI Control traffic in MCTP messages is described in subclause 9.1.1. The
311 encapsulation of NC-SI Pass-through traffic in MCTP messages is described in subclause 9.2.1.

312 # 7 NC-SI over MCTP specific considerations

313 ## 7.1 Packages and channels

314 The NC-SI specification defines different topologies using the concepts of channels and packages. A
315 channel is associated with a network port and a package is usually associated with a physical device that
316 exposes a single NC-SI bus. In an MCTP context, a package is related to an MCTP endpoint. Typically, a
317 package is identified by a single MCTP EID on an MCTP network.

318 NOTE Each device may expose multiple MCTP endpoints on different transport bindings (for example PCIe and
319 SMBus). The EID on each transport binding may be different. In this case, the NC-SI package may be associated
320 with multiple EIDs but only a single EID shall be used for NC-SI over MCTP at a given moment.

321 For example, each MCTP endpoint is associated with a PCIe endpoint and its physical address (as
322 defined in DSP0238) in an MCTP over PCIe VDM transport binding implementation. A multi-function PCIe
323 device has multiple physical addresses available. Such a device may choose to expose one NC-SI
324 package with multiple NC-SI channels via a single MCTP PCIe endpoint (as described in Figure 3) or
325 multiple NC-SI packages, each package with a single NC-SI channel exposed via a dedicated MCTP
326 PCIe endpoint (as described in Figure 4).

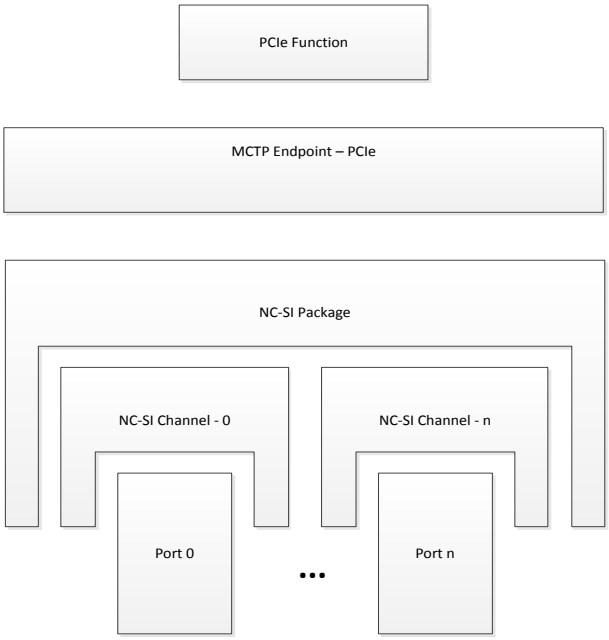

Figure 3 – Single MCTP EID to multiple NC-SI channels mapping

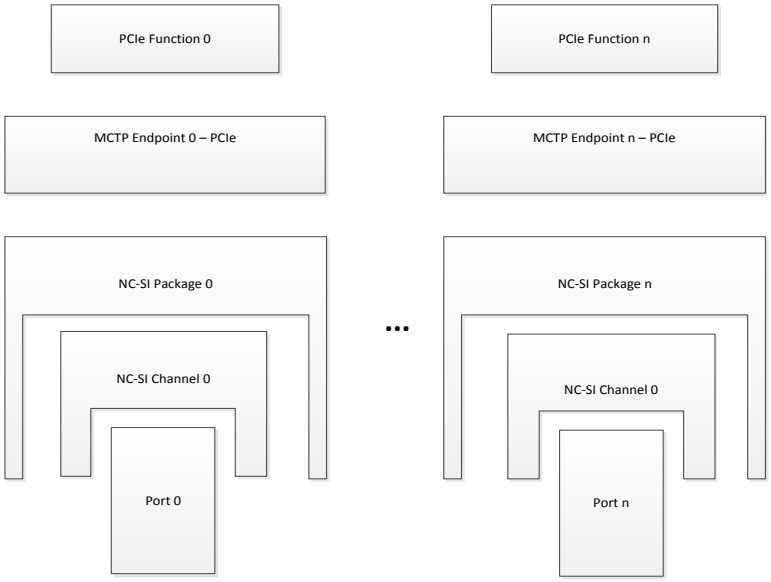

Figure 4 – Multiple MCTP EIDs to multiple NC-SI channels mapping

Multiple MCTP transport bindings handling is described in subclause 7.8.

NOTE All the MCTP message segmentation and reassembly capabilities required are defined at the package level.

7.2 Routing of NC-SI Pass-through traffic

7.2.1 Transmit NC-SI Pass-through traffic (MC to LAN)

Because multiple NC-SI channels can share an EID, identification of channel is still based on the source MAC address of the packet. Given the shared media behavior of RBT in multidrop configurations, packets

not destined to this package can be seen. In NC-SI over MCTP, the NC-SI pass-through packets are routed over an MCTP network, thus packets destined to other packages should not be expected. The NC should drop the received NC-SI TX Pass-through packets that are not destined to its package and count them as transmit errors. These errors shall be included in the “Pass-through TX Packets Dropped” counter as part of the Get NC-SI Pass-through Statistics Response.

7.2.2 Receive NC-SI Pass-through traffic (LAN to MC)

The forwarding of network traffic to the MC shall use the same rules as defined in DSP0222.

7.3 Multiple NC support

7.3.1 NC-SI arbitration support

In the original NC-SI specification, hardware and command-based arbitration are defined as ways to share an inherently point-to-point media between different NCs. With MCTP, the media itself may provide other means to arbitrate between different NCs. Thus, there is no need to use NC-SI HW arbitration method to arbitrate between multiple NCs on an MCTP network.

An NC supporting the NC-SI over MCTP binding shall retain the support for the ‘select package’ and ‘deselect package’ commands to allow control of asynchronous transmission from the NC.

7.4 Flow control

7.4.1 Flow control for MCTP packets

A physical medium supporting NC-SI over MCTP communication shall be able to buffer at least one NC-SI Control or Ethernet message at the rate of the physical layer. Flow control of MCTP packets between the Network Controller and the Management Controller (if any) may be handled by the flow control mechanisms that are specified for that particular MCTP Transport Binding for a physical medium. For example, a network controller may use the SMBus clock stretching mechanism to delay the reception of MCTP packets or may drop such packets.

7.4.2 Flow control for NC-SI over MCTP Control messages

Flow control of NC-SI Control over MCTP messages is handled by the request/response protocol used for those messages. The Network Controller shall be able to process a single NC-SI command at a time from the Management Controller. The Management Controller shall wait until getting a NC-SI response to that NC-SI command, or for a response timeout, before sending another NC-SI command over MCTP to that NC.

7.4.3 Flow control for NC-SI Pass-through packets.

The NC-SI Pass-through traffic flow control used in RBT is an Ethernet-specific technology that is not well suited to an MCTP transport. An implementation of this specification may support Ethernet flow control, but it will apply only to Ethernet messages (message type – 0x3) and not to messages of NC-SI Control over MCTP type (message type – 0x2). The method used to control the rate of transmission of Ethernet packets is beyond the scope of this specification.

7.5 Interleaving of messages

7.5.1 Interleaving of MCTP Control and NC-SI messages

According to the MCTP specification [MCTP], an endpoint shall accept MCTP Control messages that are interleaved among NC-SI Control over MCTP or Ethernet over MCTP message packets. This is to avoid

scenarios where functions such as the MCTP bus owner are 'locked out' from managing the MCTP bus because of NC-SI Pass-through traffic.

Correspondingly, MCTP Control Message responses shall be able to be interleaved among incoming NC-SI Control over MCTP or Ethernet over MCTP message packet. However, the MCTP Control Message responses may be held up and transmitted between Ethernet Messages, provided that the MCTP command request-to-response timing requirements are met.

7.5.2 Interleaving of NC-SI Control and Ethernet over MCTP messages

NC-SI Control over MCTP and Ethernet over MCTP messages to the same EID shall not be interleaved. Similar to the DSP0222 specification case, NC-SI Control and Ethernet packets are interleaved at the message level. An MC operating with multiple NC may interleave messages sent to different NCs.

7.6 Ordering rules for NC to MC traffic

The following table defines which type of messages should pass other types of packets to avoid deadlocks. The decisions are done at a message level. Interleaving within messages is defined in the previous sections. The following behaviors are expected:

- Yes–the second message (row) shall be allowed to pass the first (column) to avoid deadlock (When blocking occurs, the second message is required to pass the first message)
- Y/N–there are no requirements. The second message may optionally pass the first message or be blocked by it.
- No–the second message shall not be allowed to pass the first message. This is required to avoid out of order events.

Row Pass Column?	MCTP Control response (Col 1)	NC-SI response (Col 2)	NC-SI AEN (Col 3)	Ethernet Packet (Col 4)
MCTP Control response (Row A)	Y/N	Y/N	Yes	Y/N
NC-SI response packet (Row B)	Y/N	Y/N	Yes	Y/N
NC-SI AEN (Row C)	Y/N	Y/N	No	Y/N
Ethernet packet (Row D)	Y/N	Y/N	Y/N	No

Notes (The letter and number indicates the row and column in the table above):

- **A1** This situation will occur only in NCs accepting multiple outstanding MCTP control commands.
- **B2** This situation will occur only in NCs accepting multiple outstanding NC-SI commands.
- **A3, B3** An NC-SI AEN may be blocked if the channel is disabled or the package deselected. Thus it should not block MCTP Control or NC-SI responses.

- **C3** AENs should be sent in order of occurrence to avoid cases where the latest received status is obsolete. For example in the case of a link-down event followed by a link-up event, the AEN on the link-up event must not pass the AEN on the link-down event.

- **D4** Ethernet packets must be sent in order to avoid out-of-order events in the upper layers.

7.7 Assembly requirements

According to the interleaving requirements described in subclause 7.5, the NC shall be able to assemble a single NC-SI Control or Ethernet over MCTP message at a time. The maximum Ethernet packet size is defined in subclause 9.2. The maximum NC-SI packet size is defined in subclause 9.1.

Buffering requirements for other message types are not covered in this specification.

7.8 Multiple MCTP transport bindings

In the DSP0222 specification, the channels use a single physical interface all the time. In NC-SI over MCTP, multiple MCTP transport bindings may be used at different times to convey NC-SI traffic to allow tradeoffs between data rate and power consumption. The following requirements apply to those MCTP transport bindings:

1) NC-SI control messages (identified by MCTP message type 0x2) shall be supported

2) Ethernet messages (identified by MCTP message type 0x3) may be supported

Figure 5 shows an example of multiple MCTP transport bindings using MCTP over PCIe VDM and MCTP over SMBus. The types of NC-SI over MCTP traffic on each MCTP transport binding may vary as described above.

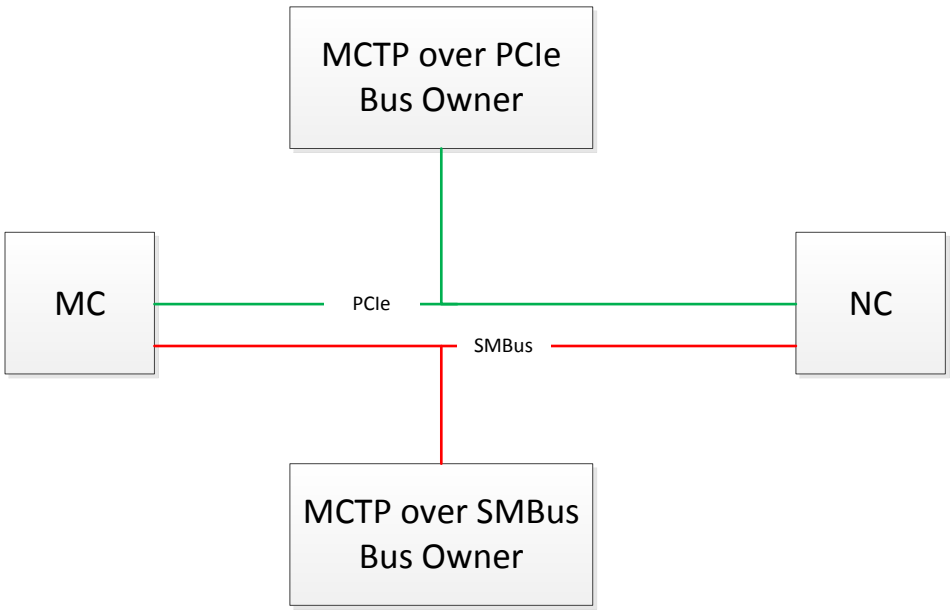

Figure 5 – Multiple MCTP transport bindings example

7.8.1 Supported message types over different MCTP transport bindings

An endpoint may support different MCTP message types over different MCTP transport bindings. For example, an NC may choose to support Ethernet message type over MCTP PCIe VDM transport only. It

is recommended that an MC initially determines the supported message types on a given medium during the discovery phase using the Get Supported Message Type MCTP Control command prior to transmitting MCTP traffic of specific MCTP message type on the medium.

7.8.2 MCTP EID and physical address changes.

The NC-SI package mapping of the NC or the MC to MCTP EID and/or physical interface address may change due to the following reasons:

1) Changes in the MCTP transport medium used. For example moving from PCIe to SMBus medium when PCIe becomes unavailable for MCTP communication due to change of power state.

2) Changes in the EID to physical address mapping. For example when changing medium or during re-enumeration process or in a multifunction PCIe device, if the function of which RID is being used is disabled by the host, the MCTP endpoint may move to another function.

In order to avoid breakup of network connections, and in order to avoid the need to reconfigure the NC, the NC-SI connection should be kept alive during the transition. The MC is responsible for the reconnection of the channel in case of address mapping changes. The next clause describes possible flows that may be used to ease the re-discovery of an NC whose address has changed. A flow by which the MC can expose a change of its own address to the NC(s) is described in subclause 7.8.4.

According to the _ specification, an MC or NC that has its physical address changed should send an MCTP Discovery Notify command to the bus owner so that the routing tables can be updated.

7.8.3 NC discovery flows

The MC may use one of the following example flows to discover a NC whose address has changed.

7.8.3.1 Full discovery

The simplest and most time consuming method is to discover the NC partner by using the standard MCTP discovery method. This method works with NCs that support at least MCTP 1.1 and NC-SI 1.0.

The following flow may be used:

- The MC detects a potential address update condition (for example: power state change, link status change, or re-enumeration) or detects an NC-SI timeout condition (as defined in section 6.8.2.1 of NC-SI).

- The MC finds all the endpoints in the system by sending an MCTP “Get Routing Table Entries” command to the bus owner and to any bridges in the MCTP network.

- For each device listed, the MC checks whether it supports the required MCTP message types (NC-SI Control and optionally Ethernet) by using the MCTP “Get Message Type Support” command.

For each potential endpoint discovered by using the method above, the MC checks whether it is the original NC partner, for example by sending an “Get Version ID” NC-SI command to the original NC ID and checking the response.

7.8.3.2 UUID based discovery

This method is based on the usage of the “Resolve UUID” MCTP command.

To use this method, the bus owner or bridge must support the “Resolve UUID” MCTP command and the NC must support the “Get Endpoint UUID” MCTP command.

The following flow may be used:

- When the NC-SI channel is first established by using some proprietary method (for example by using the flow from the previous section), the MC may send a “Get Endpoint UUID” MCTP command to the NC. It then keeps the UUID information for future use.
- MC periodically sends a “Get Routing Table” Command to the bus owner to receive updated endpoints addresses.
- The NC whose address changes or that wants to move to another active bus sends a “Discovery Notify” MCTP command to the bus owner of the new bus.
- As part of the routing table update, the bus owner sends a “Get Endpoint UUID” MCTP command to the NC and updates its routing table accordingly.
- The MC sends a “Resolve UUID” MCTP Command to the bus owner by using the previously saved NC UUID. In response, it gets the list of EIDs matching this UUID.
- The MC can check if the relevant message types (NC-SI Control and optionally Ethernet) are supported on the new bus by using an MCTP “Get Message Type Support” command.
- The MC may then send any NC-SI Command to the NC to communicate with the NC on the new medium.

7.8.3.3 NC-SI based discovery

The NC must support the “Get Supported Media” NC-SI command as defined in clause 10.1 to use this method.

The following flow may be used.

- The MC detects a potential address update condition (for example: power state change, link status change, AEN from the NC, or re-enumeration) or detects a timeout condition on NC-SI (as defined in section 6.8.2.1 of NC-SI).
- If the original bus is still available (for example, when transitioning from SMBus to PCIe), it may send on the original bus a “Get Supported Media” NC-SI command. In the response, the NC will provide information on the routing that should be used on the new bus and on the support for Pass-through on this bus.
- The MC may then send any NC-SI Command to the NC to communicate with the NC on the new medium.

This method may not be applicable when there is no active channel that can be used to send the “Get Supported Media” NC-SI command over. In this case, one of the other methods should be used.

7.8.4 MC update flow

In the case where MC physical address or its MCTP EID changes, it may send an “Enable Channel” NC-SI command to the NC. This command will update the MC EID and physical address used by the NC.

7.8.5 Transition between mediums

A transition of an NC-SI package from one medium to another can occur due to changes in the available media. For example, a transition from SMBus to PCIe can occur when PCIe becomes available to provide a larger bandwidth.

A transition of an NC-SI from one medium to another is achieved when the NC is deselected on the first medium and selected on the second medium as described in subclause 7.9.

The NC may notify the MC about the state of a medium using an AEN.

1) Potential loss of a medium prior to losing the medium
2) Availability of a new medium

Alternatively, the MC may be aware of the medium change independently, for example, by detecting its own PCIe bus became active, by interaction with the BIOS, and so on.

The MC may initiate the transition by using MCTP Control and NC-SI Control messages as described in subclause 7.8.5.1.

A transition can be between mediums with different levels of support of Ethernet MCTP messages.

When an NC transitions from a medium on which Ethernet messages were supported to a second medium on which Ethernet messages are not supported, the NC should stop sending and receiving Ethernet messages on the first medium after the NC-SI channel had been deselected on the first medium.

The MC may transition back later to the first medium for communicating Ethernet messages. If the MC transitions back to the first medium supporting Ethernet messages, it may resume communications of Ethernet messages based on the previous configuration. If the configuration was lost during the transitions, the NC shall return to the NC-SI Initial State (as described in section 6.2.4 of DSP0222).

Even if NC-SI Pass-through traffic (Ethernet messages) is supported over multiple mediums, Pass-through traffic shall not be transitioned to a new medium before the connection between the MC and the NC is re-established on the new medium. The NC shall support the following flows to initiate a transition to the new medium:

- If the current medium is still active (for example when moving from SMBus to PCIe to achieve better throughput), the NC shall keep its Pass-through traffic on the original medium (both MC to network and network to MC). The NC shall also send outstanding NC-SI responses on the original medium.

  NOTE The MC can stop the traffic from the NC on the current medium by sending “Disable Channel” and “Disable Channel Network TX” NC-SI commands to all the channels before the transition. In this case, it shall send “Enable Channel” and “Enable Channel Network TX” NC-SI commands to all active channels on the new medium, to allow the traffic to resume.

- If the current medium is inactive (for example, when moving from PCIe to SMBus due to a power transition), the NC shall stop transferring Ethernet messages. If a packet is being transmitted by the NC when the original medium becomes unavailable, the NC shall not continue the transmission of the packet and the packet may be lost. Outstanding NC-SI responses may be discarded by the NC.

- When any NC-SI command is received from the MC on the new medium (apart from “*Deselect Package*”), the NC shall move to “Selected” state on the new medium (see subclause 7.9).

  - An NC-SI Rx Pass-through message to the MC on the current medium shall be completed by the NC on the current medium and only after that shall the NC send the NC-SI response to the MC on the new medium.

  - The next Pass-through message sent to the MC after a successful response to the NC-SI command shall be sent on the new medium.

- The NC shall accept Pass-through traffic from the MC on the new medium after the NC moves to "Selected" state on the new medium and sends the first successful NC-SI response.

- The same algorithm as described above shall be used for the selection of the medium to use for sending NC-SI AEN messages to the MC.

An NC that uses multiple MCTP transport bindings should support at least one of the UUID based recovery or the NC-SI based recovery methods in addition to the Full Discovery mechanism.

7.8.5.1 MC transition flow

The MC can initiate a transition between mediums for one of the following reasons.

1) Loss of medium for NC-SI over MCTP communications. For example, system transitioning into a low power state may make PCIe medium unavailable for NC-SI over MCTP communications over PCIe VDM transport.

2) Reception of an AEN from the NC notifying a medium state change. For example, an NC may notify the MC about the potential loss of the PCIe medium, triggering a transition to SMBus.

The following flow can be used by the MC to initiate a transition between mediums:

- If the current medium is still active (for example when moving from SMbus to PCIe to achieve better throughput), the MC can keep its traffic on the original medium until it discovers the NC by using one of the flows described in subclause 7.8.3. If the current medium is inactive (for example, when moving from PCIe to SMBus due to a power transition), the MC will stop transferring Ethernet messages with NC until discovery of the NC.

- The MC can then send an “Enable Channel” NC-SI Command, or any other command to the NC to select it on the new medium. The MC will then wait for the NC response before starting to send packets on the new medium. The MC will complete transmission of the current Ethernet message before sending the command and will not send Ethernet messages while waiting for the response. The MC will accept Ethernet message on the original medium until the response from the NC is received on the current medium.

- If Pass-through is supported by the NC over only a single medium, when transitioning out of this medium, the MC will not send Pass-through traffic to the NC and will not expect to receive traffic from the NC.

- If a medium becomes unavailable while an MC waits for an NC-SI command response, it can assume the command was lost and retry it on the new medium.

7.9 Package selection

The “Selected” state of an NC-SI package is defined for each of the MCTP transports to which it can bind. A package can be selected only on a single MCTP medium at a given point of time.

As in DSP0222, a package is selected by reception of a “Select Package” on the MCTP medium or any other command except “Deselect Package”.

A package is deselected on a specific MCTP medium by reception of a “Deselect Package” command, selection of the package on another medium or if the physical medium on which it operate becomes unavailable. If the packet is deselected by an NC-SI command it should move to the deselected state only after sending a response to the command.

A package is allowed to send Ethernet messages or NC-SI Control messages on an MCTP medium only if in the “Selected” state on that medium.

An NC should use the source EID and source physical address received from the last received NC-SI command to respond to this command and as the destination of subsequent Ethernet messages. If a command is received during the transmission of an Ethernet message, the destination should change only at the beginning of the next message.

The channel selection state and all other NC-SI configurations may be kept during the transition from one medium to another. If the configuration is altered during the transition, the NC shall return to Initial State.

8 Supported NC-SI commands

The supported NC-SI commands when bound to MCTP is a subset of the commands in DSP0222 specification. The subset of supported commands varies according to the supported messages as indicated in the response to the Get Message Type Support MCTP Control command. If only the NC-SI Control message type is supported, the commands related to the Pass-through traffic control are not supported. If both the NC-SI Control and Ethernet message types are supported, these commands are supported. Table 2 lists the supported commands according to the supported message types.

Optional commands may have different implementation over different media.

Table 2 – Supported NC-SI commands

Command Type	Command Name	Description	Response Type	Command Support Requirement NC-SI Control Messages Only	Command Support Requirement NC-SI Control and Ethernet Messages
0x00	Clear Initial State	Used by the Management Controller to acknowledge that the Network Controller is in the Initial State	0x80	M	M
0x01	Select Package	Used to explicitly select a controller package to transmit packets through the NC-SI interface	0x81	O3	O[3]
0x02	Deselect Package	Used to explicitly instruct the controller package to stop transmitting packets through the NC-SI interface	0x82	O[3]	O[3]
0x03	Enable Channel	Used to enable the NC-SI channel and to start the forwarding of bidirectional Management Controller packets	0x83	M	M
0x04	Disable Channel	Used to disable the NC-SI channel and to stop the forwarding of bidirectional Management Controller packets	0x84	M	M
0x05	Reset Channel	Used to synchronously put the Network Controller back to the Initial State	0x85	M	M
0x06	Enable Channel Network TX	Used to explicitly enable the channel to transmit Pass-through packets onto the network	0x86	N/A	M

Command Type	Command Name	Description	Response Type	Command Support Requirement NC-SI Control Messages Only	Command Support Requirement NC-SI Control and Ethernet Messages
0x07	Disable Channel Network TX	Used to explicitly disable the channel from transmitting Pass-through packets onto the network	0x87	N/A	M
0x08	AEN Enable	Used to control generating AENs	0x88	C	C
0x09	Set Link	Used during OS absence to force link settings, or to return to auto-negotiation mode	0x89	O	M
0x0A	Get Link Status	Used to get current link status information	0x8A	O	M
0x0B	Set VLAN Filter	Used to program VLAN IDs for VLAN filtering	0x8B	N/A	M
0x0C	Enable VLAN	Used to enable VLAN filtering of Management Controller RX packets	0x8C	N/A	M
0x0D	Disable VLAN	Used to disable VLAN filtering	0x8D	N/A	M
0x0E	Set MAC Address	Used to configure and enable unicast and multicast MAC address filters	0x8E	N/A	M
0x10	Enable Broadcast Filtering	Used to enable full or selective broadcast packet filtering	0x90	N/A	M
0x11	Disable Broadcast Filtering	Used to disable all broadcast packet filtering, and to enable the forwarding of broadcast packets	0x91	N/A	M
0x12	Enable Global Multicast Filtering	Used to disable forwarding of all multicast packets to the Management Controller	0x92	N/A	C
0x13	Disable Global Multicast Filtering	Used to enable forwarding of all multicast packets to the Management Controller	0x93	N/A	C
0x14	Set NC-SI Flow Control	Used to configure IEEE 802.3 flow control on NC-SI	0x94	N/A	O
0x15	Get Version ID	Used to get controller-related version information	0x95	M	M
0x16	Get Capabilities	Used to get optional functions supported by the NC	0x96	M[1]	M
0x17	Get Parameters	Used to get configuration parameter values currently in effect on the controller	0x97	M2	M

Command Type	Command Name	Description	Response Type	Command Support Requirement NC-SI Control Messages Only	Command Support Requirement NC-SI Control and Ethernet Messages
0x18	Get Controller Packet Statistics	Used to get current packet statistics for the Network Controller	0x98	O	O
0x19	Get NC-SI Statistics	Used to request the packet statistics specific to the NC-SI interface	0x99	O	O
0x1A	Get NC-SI Pass-through Statistics	Used to request NC-SI Pass-through packet statistics	0x9A	N/A	O
0x1B	Get Supported Media	Used to return the media on which NC-SI can run and routing information for each medium.	0x9B	O	O
0x50	OEM Command	Used to request vendor-specific data	0xD0	O	O

Key: M = Mandatory (required)
O = Optional
C = Conditional (see command description)
N/A = Not applicable

1. The only part of the response that is relevant is the AEN control support field.
2. The only part of the response that is relevant is the Link Settings, AEN control fields and the Channel Enabled flag in the Configuration Flags.
3. The 'Select Package' and 'Deselect Package' commands impact only transmission of NC-SI Control and Ethernet over MCTP message types and do not impact other MCTP message types.

9 Message types

The MC and the NC shall support the NC-SI over MCTP message type (0x02). The Ethernet message type should be supported if NC-SI Pass-through traffic is expected on this interface.

9.1 NC-SI message type (0x02)

This message type is used to carry NC-SI Control packets that are identified by the NC-SI Ethertype in the DSP0222 specification. This includes command, response, and AEN packets.

The maximum NC-SI message payload size is 1500 bytes to keep the same limit as in NC-SI. This includes the payload starting from the MC ID field.

9.1.1 Encapsulation

The encapsulation of NC-SI Control packets includes the packet as described in the Control packet data structure of DSP0222 specification encapsulated in an MCTP header. NC-SI messages may be fragmented to multiple MCTP packets.

NC-SI control packets communicated over MCTP do not follow the Ethernet frame encapsulation defined in DSP0222 for NC-SI over RMII Based Transport (RBT) transport binding. NC-SI control packets over MCTP shall not include Ethernet frame header, Ethernet packet pad, and Ethernet Frame Check

Sequence (FCS). Instead, the encapsulation described in Table 3 shall be used to encapsulate NC-SI control messages.

NOTE The Control packets frames in DSP0222 uses a DA, SA, and Ethertype MAC header. The DA and SA part do not contain any useful data and the Ethertype is used to differentiate between Control packets and Ethernet traffic. In NC-SI over MCTP, this Ethernet framing is not used, as the differentiation is achieved through usage of different message types.

Table 3 –NC-SI messages encapsulation

Bytes	+0								+1								+2								+3							
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
00..03	RSVD				Header Version				Destination Endpoint ID								Source Endpoint ID								SOM	EOM	Pkt seq #			TO	Message Tag		
04..07	IC 0	Message Type 0x02							MC ID								Header Revision								Reserved							
08..11	IID								Command								Channel ID								Reserved						Payload Len	
12..15	Payload Length								Reserved																							
16..19	Reserved																															
20..23	Reserved								Control Packet Payload																							
...	...																															
...	...								Control Packet Payload								Payload Padding (as required)															
...	Payload Padding (as rqrd)								Checksum 3:1																							
...	Checksum 0																															

See NC-SI for details of the NC-SI Control packets format.

The following tables describe the value for the various fields of the message whose description differs from the description in the MCTP or NC-SI specification.

Table 4 - MCTP Transport Header fields

Field Name	Field Size	Value	Comment
Tag Owner (TO)	1 bit	Varies	Indicates that the Tag field value was generated by the message source = 0b Tag not from message source = 1b Tag from message source. Should be set for Commands and AEN packets. Should be cleared for Response packets.
Msg Tag	3 bits	Varies	The Tag field shall be set by the source of the message.

640 **Table 5 – MCTP Specific Message Header field**

Field Name	Field Size	Value	Comment
IC Bit	1 bit	0b	NC-SI over MCTP does not define message integrity check as it relies on the MCTP packet integrity check provided by the underlying medium or checks that are encapsulated in the message payload. This field is present only in the first packet of a message (SOM = 1).
Message Type	7 bits	0x02	Identifies the MCTP message type as an NC-SI Control over MCTP message. This field is present only in the first packet of a message (SOM = 1).

641 ### 9.1.2 Version

642 The version that shall be reported for this message type in the Get MCTP Version Support response is
643 0xF1F0F000 (1.0.0)

644 ## 9.2 Ethernet message type (0x03)

645 This message type is used to carry NC-SI Pass-through packets. Ethernet messages may be fragmented
646 to multiple MCTP packets.

647 The nominal Ethernet message size that shall be supported is 1518 bytes to accommodate a full Ethernet
648 packet including a VLAN but without FCS. If additional L2 tags are expected (for example, MACSec), the
649 supported packet size shall increase accordingly.

650 ### 9.2.1 Encapsulation

651 The encapsulation of Ethernet packets includes the entire Ethernet frame from the Source MAC address
652 to the end of the payload, not including the FCS, prefixed with an MCTP header.

653 NOTE In NC-SI, the FCS was required as part of the Ethernet encapsulation used over RMII. When Ethernet
654 packets are sent over other mediums, the medium specific error recovery mechanisms are used and the FCS is not
655 required.

656 The FCS should be added by the NC for packets sent by the MC to the network and should be checked
657 and removed by the NC for packets received from the network to the MC. Packets with a wrong FCS
658 should not be forwarded to the MC.

659 This behavior is consistent with the FCS offload provided by NCs to the host OS.

Table 6 – Ethernet messages encapsulation

Bytes	+0								+1								+2								+3							
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
00..03	RSVD				Header Version				Destination Endpoint ID								Source Endpoint ID								SOM	EOM	Pkt seq #			TO	Message Tag			
04..07	IC 0				Message Type 0x03				Destination Address5:3																								
08..11	Destination Address 2:0																								Source Address 5							
12..15	Source Address 4:1																															
16..	Source Address 0								Optional L2 tags																							
...	Optional L2 tags								Ethertype																Ethernet Payload							
...	Ethernet Payload (no FCS)																															

The following tables describe the value for the various fields of the message whose description differs from the description in the MCTP or NC-SI specification.

Table 7 - MCTP Transport Header fields

Field Name	Field Size	Value	Comment
Tag Owner (TO)	1 bit	1b	Indicates that the Tag field value was generated by the message source = 0b Tag not from message source = 1b Tag from message source Should be set for all packets
Msg Tag	3 bits	Varies	The Tag field shall be set by the source of the message.

Table 8 – MCTP Specific Message Header field

Field Name	Field Size	Value	Comment
IC Bit	1 bit	0b	NC-SI over MCTP does not define a message integrity check because it relies on the MCTP packet integrity check provided by the underlying medium or checks that are encapsulated in the message payload. This field is present only in the first packet of a message (SOM = 1).
Message Type	7 bits	0x03	Identifies the MCTP message type as an Ethernet over MCTP message. This field is present only in the first packet of a message (SOM = 1).

9.2.2 Version

The version that shall be reported for this message type in the Get MCTP Version Support response is 0xF1F0F000 (1.0.0)

10 NC-SI support specific to MCTP transport

The following command and AEN may be implemented as part of this specification to allow an implementation of the discovery flow described in clause 7.8.3.3.

10.1 Get Supported Media command (0x51)

This command is used to query a device about the Media on which NC-SI can be conveyed. This command is optional and is applicable only if more than one media is supported.

The Get Supported Media command is addressed to the package, rather than to a particular channel (that is, the command is sent with a Channel ID where the Package ID subfield matches the ID of the intended package and the Internal Channel ID subfield is set to `0x1F`).

Table 9 illustrates the packet format of the Get Supported Media command.

Table 9 – Get Supported Media command packet format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Checksum			
20..45	Pad			

10.2 Get Supported Media response (0xD1)

In the absence of any error, the package shall process and respond to the Get Supported Media command by sending the response packet and payload shown in Table 10.

Table 10 – Get Supported Media response packet format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Reserved			Number of medias supported
20..	Media descriptors as described in Table 11. The number of media descriptors is according to the Number of medias supported field value.			
…	Checksum			
…	Pad			

685 **Table 11 – Get Supported Media response media descriptors format**

Byte	Description
0	EID. Should be 0x0 if Physical Medium Identifier is RBT.
1	Physical Transport Binding Identifier, according to MCTP ID specification (DSP0239). Should be 0x0 if Physical Medium Type Identifier is RBT.
2	Physical Medium Identifier, according to MCTP ID specification (DSP0239). This value is used to indicate what format the following physical address data is given in.
3.0	NC-SI Pass-through is supported. 0: NC-SI Pass-through is not supported over this medium. 1: NC-SI Pass-through is supported over this medium.
3.6:1	Reserved
3.7	Status 0: Medium is not available. 1: Medium is available.
4	Physical Address Size. Should be 0x0 if Physical Medium Identifier is RBT.
5:N	Physical Address. This field is not present if Physical Medium Identifier is RBT. If present, this field is valid only if the Status bit is set and its value is unspecified otherwise.

686 ## 10.3 Medium change AEN

687 The Medium change AEN is used to alert the MC that there was a status change in one of the media
688 supported by the NC, or such a change is expected according to some external or internal condition
689 detected by the NC.

690 This AEN should be sent if any change occurred in the status of one of the media supported by the
691 device. It may also be sent for expected changes in the medium status, if the NC is aware of them.

692 For example, if while NC-SI package is active over SMBus, the PCIe bus becomes available, this AEN
693 should be sent. Another example, if while NC-SI package is active over PCIe, the NC detects that the
694 PCIe bus is going to be disabled, it may send this AEN also.

695 The media descriptors field reproduces the bit definitions defined in the Get Supported Media response
696 (Table 11).

697 **Table 12 – Medium change AEN format**

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Reserved			AEN Type = 0x70
20..23	Reserved			Number of Medias supported.
24..	Media descriptors			
…	Checksum			
…	Pad			

698 # 11 Packet-Based Timing specific to MCTP Binding

699 Table 13 presents changes in the NC-SI timing parameters relative to NC-SI Packet-Based and Op-Code
700 Timing Parameters Table in DSP0222. Parameters not listed in the table below should be taken from the
701 table in DSP0222.

702 **Table 13 – NC-SI Timing Parameters specific to MCTP Binding**

Name	Symbol	Value	Description
Normal Execution Interval	T5	50 ms, max	Maximum time interval from when a controller receives a command to when it delivers a response to that command, unless otherwise specified Measured from the rising edge of the first clock following the last bit of the command packet to the rising edge of the clock for the first bit of the response packet Note: When T5 passed, an extension of the timeout should be allowed and taken into consideration under the following conditions: 1. An Ethernet message or an NC-SI control message (AEN) being transmitted, 2. On a shared medium, the medium is occupied by other devices.

703

ANNEX A
(informative)
Notation and conventions

A.1 Notations

Examples of notations used in this document are as follows:

- 2:N — In field descriptions, this will typically be used to represent a range of byte offsets starting from byte two and continuing to and including byte N. The lowest offset is on the left; the highest is on the right.
- (6) — Parentheses around a single number can be used in message field descriptions to indicate a byte field that may be present or absent.
- (3:6) — Parentheses around a field consisting of a range of bytes indicates the entire range may be present or absent. The lowest offset is on the left; the highest is on the right.
- PCIe — Underlined, blue text is typically used to indicate a reference to a document or specification called out in "Normative references" clause or to items hyperlinked within the document.
- rsvd — This case-insensitive abbreviation is for “reserved.”
- [4] — Square brackets around a number are typically used to indicate a bit offset. Bit offsets are given as zero-based values (that is, the least significant bit [LSb] offset = 0).
- [7:5] — This notation indicates a range of bit offsets. The most significant bit is on the left; the least significant bit is on the right.
- `1b` — The lowercase "`b`" following a number consisting of `0`s and `1`s is used to indicate the number is being given in binary format.
- `0x12A` — A leading "`0x`" is used to indicate a number given in hexadecimal format.

728 # ANNEX B
729 (informative)
730
731 # Change log

Version	Date	Description
1.0.0	2013-08-22	
1.1.0	2015-03-21	Typos: • Fixed wrong message type in Table 6 Functional changes: • Stronger requirement on NC-SI control messages encapsulation. • Added specific timing requirements. • Added ability to send AEN on upcoming media status changes.

732