



1
2
3
4

Document Number: DSP0238

Date: 28 July 2009

Version: 1.0.0

5 **Management Component Transport Protocol**
6 **(MCTP) PCIe VDM Transport Binding**
7 **Specification**

8 **Document Type: Specification**
9 **Document Status: DMTF Standard**
10 **Document Language: E**
11

12 Copyright Notice

13 Copyright © 2009 Distributed Management Task Force, Inc. (DMTF). All rights reserved.

14 DMTF is a not-for-profit association of industry members dedicated to promoting enterprise and systems
15 management and interoperability. Members and non-members may reproduce DMTF specifications and
16 documents, provided that correct attribution is given. As DMTF specifications may be revised from time to
17 time, the particular version and release date should always be noted.

18 Implementation of certain elements of this standard or proposed standard may be subject to third party
19 patent rights, including provisional patent rights (herein "patent rights"). DMTF makes no representations
20 to users of the standard as to the existence of such rights, and is not responsible to recognize, disclose,
21 or identify any or all such third party patent right, owners or claimants, nor for any incomplete or
22 inaccurate identification or disclosure of such rights, owners or claimants. DMTF shall have no liability to
23 any party, in any manner or circumstance, under any legal theory whatsoever, for failure to recognize,
24 disclose, or identify any such third party patent rights, or for such party's reliance on the standard or
25 incorporation thereof in its product, protocols or testing procedures. DMTF shall have no liability to any
26 party implementing such standard, whether such implementation is foreseeable or not, nor to any patent
27 owner or claimant, and shall have no liability or responsibility for costs or losses incurred if a standard is
28 withdrawn or modified after publication, and shall be indemnified and held harmless by any party
29 implementing the standard from any and all claims of infringement by a patent owner for such
30 implementations.

31 For information about patents held by third-parties which have notified the DMTF that, in their opinion,
32 such patent may relate to or impact implementations of DMTF standards, visit
33 <http://www.dmtf.org/about/policies/disclosures.php>.

34 PCI-SIG, PCIe, and the PCI HOT PLUG design mark are registered trademarks or service marks of PCI-
35 SIG.

36 All other marks and brands are the property of their respective owners.

37

38

CONTENTS

39 Foreword 4

40 Introduction 5

41 1 Scope 7

42 2 Normative References..... 7

43 2.1 Approved References 7

44 2.2 Other References..... 7

45 3 Terms and Definitions..... 7

46 4 Symbols and Abbreviated Terms..... 8

47 5 Conventions 8

48 5.1 Reserved and Unassigned Values 8

49 5.2 Byte Ordering..... 8

50 6 MCTP over PCI Express VDM Transport..... 8

51 6.1 Packet Format..... 8

52 6.2 Supported Media..... 10

53 6.3 Physical Address Format for MCTP Control Messages 11

54 6.4 Message Routing 11

55 6.5 Bus Owner Address 12

56 6.6 Bus Address Assignment for PCIe 12

57 6.7 Host Dependencies..... 12

58 6.8 Discovery Notify Message Use for PCIe..... 13

59 6.9 MCTP over PCIe Endpoint Discovery..... 13

60 6.10 MCTP Messages Timing Requirements 17

61 Annex A (informative) Notations and Conventions 18

62 Annex B (informative) Change Log..... 19

63

64 Figures

65 Figure 1 – MCTP over PCI Express Packet Format 9

66 Figure 2 – Flow of Operations for Full PCIe MCTP Discovery 15

67 Figure 3 – Flow of Operations for Partial Endpoint Discovery 16

68

69 Tables

70 Table 1 – PCI Express Medium-Specific MCTP Packet Fields 9

71 Table 2 – Supported Media..... 10

72 Table 3 – Physical Address Format 11

73 Table 4 – Timing Specifications for MCTP Control Messages on PCIe VDM 17

74

75

Foreword

76 The *Management Component Transport Protocol (MCTP) PCIe VDM Transport Binding Specification*
77 (DSP0238) was prepared by the PMCI Working Group.

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

80

Introduction

81 The Management Component Transport Protocol (MCTP) defines a communication model intended to
82 facilitate communication between:

- 83 • Management controllers and other management controllers
- 84 • Management controllers and management devices

85 The communication model includes a message format, transport description, message exchange
86 patterns, and configuration and initialization messages.

87 The [MCTP Base Specification](#) describes the protocol and commands used for communication within and
88 initialization of an MCTP network. Associated with the [MCTP Base Specification](#) are transport binding
89 specifications that define how the MCTP base protocol and MCTP control commands are implemented on
90 a particular physical transport type and medium, such as SMBus/I²C, PCI Express™ (PCIe) Vendor
91 Defined Messaging (VDM), and so on.

92

93 Management Component Transport Protocol (MCTP) PCIe 94 VDM Transport Binding Specification

95 1 Scope

96 This document provides the specifications for the Management Component Transport Protocol (MCTP)
97 transport binding for PCI Express™ using PCIe Vendor Defined Messages (VDMs).

98 2 Normative References

99 The following referenced documents are indispensable for the application of this document. For dated
100 references, only the edition cited applies. For undated references, the latest edition of the referenced
101 document (including any amendments) applies.

102 2.1 Approved References

103 DMTF DSP0236, *Management Component Transport Protocol (MCTP) Base Specification 1.0*, MCTP,
104 http://www.dmtf.org/standards/published_documents/DSP0236_1.0.pdf

105 DMTF DSP0239, *Management Component Transport Protocol (MCTP) IDs and Codes 1.0*, MCTP_ID,
106 http://www.dmtf.org/standards/published_documents/DSP0239_1.0.pdf

107 2.2 Other References

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

110 PCI-SIG, *PCI Express Base Specification 1.1*, PCIeV1.1, March 28, 2005,
111 http://www.pcisig.com/members/downloads/specifications/pciexpress/PCI_Express_Base_1.1.pdf

112 PCI-SIG, *PCI Express Base Specification 2.0*, PCIeV2.0, December 20, 2006,
113 http://www.pcisig.com/members/downloads/specifications/pciexpress/PCI_Express_Base_2.0.pdf

114 3 Terms and Definitions

115 Refer to [DSP0236](#) for terms and definitions that are used across the MCTP specifications. For the
116 purposes of this document, the following additional terms and definitions apply.

117 3.1

118 MCTP PCIe Endpoint

119 a PCIe endpoint on which MCTP PCIe VDM communication is supported

120

121

122 4 Symbols and Abbreviated Terms

123 Refer to [DSP0236](#) for symbols and abbreviated terms that are used across the MCTP specifications. The
124 following symbols and abbreviations are used in this document.

125 4.1

126 **PCIe®**

127 PCI Express™

128 4.2

129 **VDM**

130 Vendor Defined Message

131 5 Conventions

132 The conventions described in the following clauses apply to this specification.

133 5.1 Reserved and Unassigned Values

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

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

138 5.2 Byte Ordering

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

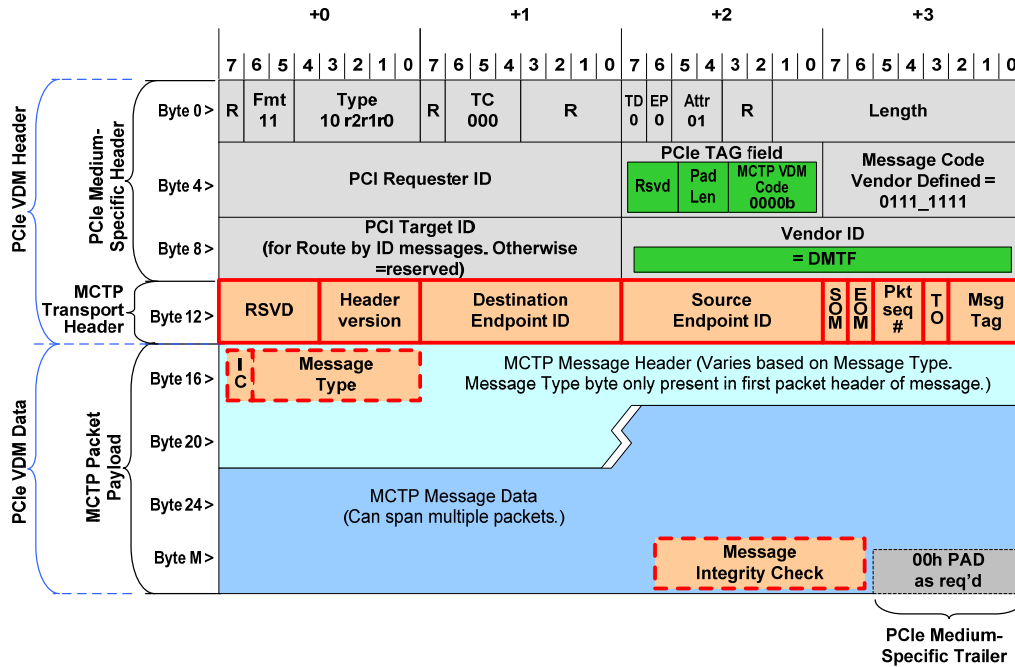
141 6 MCTP over PCI Express VDM Transport

142 This document defines the medium-specific transport binding for transferring MCTP packets between
143 endpoints on PCI Express™ using PCIe Vendor Defined Messages (VDMs).

144 6.1 Packet Format

145 The MCTP over PCI Express (PCIe) VDM transport binding transfers MCTP messages using PCIe Type
146 1 VDMs with data. MCTP messages use the MCTP VDM code value (0000b) that uniquely differentiates
147 MCTP messages from other DMTF VDMs.

148 Figure 1 shows the encapsulation of MCTP packet fields within a PCIe VDM for PCIe 1.1.



149

150

Figure 1 – MCTP over PCI Express Packet Format

151 The fields labeled “PCIe Medium-Specific Header” and “PCIe Medium-Specific Trailer” are specific to
 152 carrying MCTP packets using PCIe VDMs. The fields labeled “MCTP Transport Header” and “MCTP
 153 Packet Payload” are common fields for all MCTP packets and messages and are specified in [MCTP](#). This
 154 document defines the location of those fields when they are carried in a PCIe VDM. This document also
 155 specifies the *medium-specific* use of the MCTP “Hdr Version” field.

156 Table 1 lists the PCIe medium-specific fields and field values.

157

Table 1 – PCI Express Medium-Specific MCTP Packet Fields

Field	Description
R or Fmt[2]	PCIe 1.1/2.0: PCIe reserved bit (1 bit). PCIe 2.1: Fmt[2]. Set to 0b.
Fmt	Format (2 bits). Set to 11b to indicate 4 dword header with data.
Type	Type and Routing (5 bits). [4:3] Set to 10b to indicate a message [2:0] PCI message routing (r2r1r0) 000b : Route to Root Complex 010b : Route by ID 011b : Broadcast from Root Complex Other routing fields values are not supported for MCTP.
R	PCIe reserved bits (1 bit). Refer to the PCI Express™ specification (PCIe).
TC	Traffic Class (3 bits). Set to 000b for all MCTP over PCIe VDM.
R or R Attr R TH	PCIe 1.1/2.0: PCIe reserved bits (4 bits). PCIe 2.1: PCIe reserved bit (1 bit), Attr[2] (1 bit) – Set to 0b, reserved bit (1bit), and TH (1bit) – Set to 0b.

Field	Description
TD	TLP Digest (1 bit). Set to 0b for all MCTP over PCIe VDM.
EP	Error Present (1 bit). Set to 0b for all MCTP over PCIe VDM.
Attr	Attributes (2 bits). Set to 00b or 01b for all MCTP over PCIe VDM.
R or AT	PCIe 1.1: PCIe reserved bits (2 bits). PCIe 2.0/2.1: Address Type (AT) field. Set to 00b.
Length	Length: Length of the data in dwords. Implementations shall support the baseline MTU defined in the MCTP Base Specification . An implementation may optionally support larger transfer unit sizes. See the MCTP Base Specification for more information. (For the baseline transmission unit, legal values can be between 1–16 dwords [1 byte to 64 bytes]).
PCI Requester ID	Bus/device/function number of the managed endpoint sending the message.
Pad Len	Pad Length (2-bits). 1-based count (0 to 3) of the number of 0x00 pad bytes that have been added to the end of the packet to make the packet dword aligned with respect to PCIe . Because only packets with the EOM bit set to 1b are allowed to be less than the transfer unit size, packets that have the EOM bit set to 0b will already be dword aligned and will thus not require any pad bytes and will have a pad length of 00b.
MCTP VDM Code	Value that uniquely differentiates MCTP messages from other DMTF VDMs. Set to 0000b for this transport mapping as defined in this specification.
Message Code	(8 bits). Set to 0111_1111b to indicate a Type 1 VDM.
PCI Target ID	(16 bits). For Route By ID messages, this is the bus/device/function number that is the physical address of the target endpoint. This field is ignored for Broadcast and for Route to Root Complex messages.
Vendor ID	(16 bits). Set to 6836 (0x1AB4) for DMTF VDMs. The most significant byte is in byte 10, the least significant byte is byte 11.
Rsvd	MCTP reserved (4 bits). Set these bits to 0 when generating a message. Ignore them on incoming messages.
Hdr Version	MCTP version (4 bits) 0001b : For MCTP devices that conform to the MCTP Base Specification and this version of the PCIe VDM transport binding. All other settings: Reserved to support future packet header field expansion or header version.
0x00 PAD	Pad bytes. 0 to 3 bytes of 0x00 as required to fill out the overall PCIe VDM data to be an integral number of dwords. Because only packets with the EOM bit set to 1b are allowed to be less than the transfer unit size, packets that have the EOM bit set to 0b will already be dword aligned, and will thus not require any pad bytes and will have a pad length of 00b.

158 6.2 Supported Media

159 This physical transport binding has been designed to work with the following media specified in Table 2.
160 Use of this binding with other types of physical media is not covered by this specification.

161

Table 2 – Supported Media

Physical Media Identifier	Description
0x08	PCIe 1.1 compatible
0x09	PCIe 2.0 compatible

0x0A	PCIe 2.1 compatible
------	---------------------

162 **6.3 Physical Address Format for MCTP Control Messages**

163 The address format shown in Table 3 is used for MCTP control commands that require a physical
 164 address parameter to be returned for a bus that uses this transport binding with one of the supported
 165 media types listed in 6.2. This includes commands such as the Resolve Endpoint ID, Routing Information
 166 Update, and Get Routing Table Entries commands.

167 **Table 3 – Physical Address Format**

Format Size	Layout and Description	
2 bytes	byte 1	[7:0] – Bus number
	byte 2	[7:3] – Device number [2:0] – Function number

168 **6.4 Message Routing**

169 Physical packet routing within a PCIe bus uses routing as defined by the PCIe specification. PCIe
 170 physical routing/bridging is not the same thing as MCTP bridging. PCIe physical routing/bridging is
 171 generally transparent to MCTP. There are no MCTP-defined functions for configuring or controlling the
 172 setup of a PCIe bus. The following types of PCIe addressing are used with MCTP messages:

173 • **Route by ID**

174 All MCTP over PCIe messages between endpoints that are not the bus owner shall use Route
 175 by ID for message routing.

176 The bus owner also can use Route by ID for messages to individual endpoints.

177 PCIe endpoints are required to capture the PCIe requester ID and the MCTP source EID when
 178 receiving an EID assignment request message. This is because this command can only be
 179 issued by the PCIe bus owner.

180 • **Route to Root Complex**

181 Endpoints shall use this routing for the Discovery Notify request message to the bus owner as
 182 part of the MCTP over PCIe discovery process.

183 The PCIe endpoints shall use this routing for responding to the request messages that were
 184 sent using Broadcast from Root Complex.

185 • **Broadcast from Root Complex**

186 The MCTP PCIe bus owner should use this routing for the Prepare for Endpoint Discovery and
 187 Endpoint Discovery messages as part of the MCTP over PCIe discovery process.

188 **6.4.1 Routing Peer Transactions**

189 Because the PCIe specification does not require peer support in root complexes, MCTP over PCIe
 190 messages are not required to be routed to peer devices directly. In this case, all messages between two
 191 MCTP endpoints shall be routed to or through the PCIe bus owner as an MCTP bridge. If the PCIe bus
 192 does support peer-to-peer routing, the bus owner can support the use of direct physical addressing
 193 between endpoints.

194 6.4.2 Routing Messages between PCIe and Other Buses

195 All MCTP messages that span between PCIe and other buses shall be sent through the PCIe bus owner.
196 The PCIe bus owner has the destination EID routing tables necessary to route messages between the
197 two bus segments.

198 If an endpoint is aware of multiple routes to a destination over multiple bus types, a higher level
199 algorithm/protocol above MCTP shall be used to determine which bus/route to use. Typically this decision
200 can be based on things like power state and MCTP discovery state.

201 6.5 Bus Owner Address

202 The PCIe VDM bus owner functionality shall be accessible through "Route-to-Root Complex" addressing.

203 6.6 Bus Address Assignment for PCIe

204 PCIe bus addresses are assigned per the mechanisms specified in [PCIe](#).

205 6.7 Host Dependencies

206 MCTP over PCIe VDM, when used in a typical "PC" computer system, has a dependency on the host
207 CPU, host software, power management states, link states, and reset. Some of these dependencies are
208 described as follows:

- 209 • **Reset**

210 Assertion of "Fundamental Reset" on the bus causes both the host functionality as well as the
211 manageability portion of an MCTP PCIe endpoint to be reset. From the assertion "Fundamental
212 Reset" until the PCIe fabric has been configured and enumerated, no "MCTP over PCI Express"
213 messages can be sent.

214 Similarly, if MCTP PCI-e VDM communication is supported on a function, a function level reset
215 (FLR) could reset MCTP PCIe endpoint as well as MCTP PCIe VDM communication on that
216 function.

- 217 • **Configuration and Enumeration**

218 Following the de-assertion "Fundamental Reset", the software running on the host CPU
219 configures and enumerates the PCIe fabric. Failure of the host CPU or boot software to properly
220 configure and enumerate the PCIe fabric prevents it from being used for MCTP messaging.

- 221 • **Power Management States**

222 The host (as defined in the context of the [PCI Express™ specification](#)) controls PCIe bus power
223 management. The host may power down PCIe devices and links, or place them in sleep states,
224 independent of management controllers, which may cause MCTP PCIe VDM communication to
225 be unavailable. Depending on the device usage in the system, a PCIe device may retain or lose
226 states such as EID, "discovered" state, and routing information (if the device is a bridge). A
227 PCIe device that loses MCTP PCIe VDM communication state needs to be reinitialized and/or
228 rediscovered after it returns to a power state that supports MCTP communication.

- 229 • **Link States**

230 The PCIe link states affect MCTP over PCIe VDM communications. MCTP communication can
231 be performed only when the PCIe link is in a state that allows VDM communications. The
232 mechanisms for PCIe link state transitions are outside the scope of this specification.

233 6.8 Discovery Notify Message Use for PCIe

234 An MCTP control Discovery Notify message shall be sent from a PCIe endpoint to the PCIe bus owner
235 whenever the physical address for the device changes (that is, the endpoint receives a Type 0
236 configuration write request and the bus number is different than the currently stored bus number). This
237 occurs on the first Type 0 configuration write following a PCIe bus reset during initial enumeration, or
238 during re-enumeration where the bus number has changed (for example, because of a hot plug event,
239 bus reset, and so on).

240 Endpoints use the Discovery Notify command to inform the bus owner that it needs to update the
241 endpoint's ID. The Discovery Notify command shall be sent with the PCIe message routing set to 000b
242 (Route-to-Root Complex), the Destination Endpoint ID for the Discovery Notify message shall be set to
243 the Null Destination EID. The Source Endpoint ID field shall be set to the Null Source EID if the device
244 has not yet been assigned an EID; otherwise, it shall contain the assigned EID value.

245 6.9 MCTP over PCIe Endpoint Discovery

246 This clause describes the steps used to support discovering MCTP endpoints on PCIe.

247 6.9.1 Discovered Flag

248 Each endpoint (except the bus owner) on the PCIe bus maintains an internal flag called the *Discovered*
249 flag.

250 The flag is set to the *discovered* state when the Set Endpoint ID command is received.

251 The Prepare for Endpoint Discovery message causes each recipient endpoint on the PCIe bus to set their
252 respective Discovered flag to the *undiscovered* state. For the Prepare for Endpoint Discovery request
253 message, the routing in the physical transport header should be set to 011b (Broadcast from Root
254 Complex).

255 An endpoint also sets the flag to the *undiscovered* state at the following times:

- 256 • Whenever the PCI bus/device/function number associated with the endpoint is initially assigned
257 or is changed to a different value.
- 258 • Whenever an endpoint first appears on the bus and requires an EID assignment. A device shall
259 have been enumerated on PCI and have a bus/device/function number before it can do this.
- 260 • During operation if an endpoint enters a state that causes it to lose its EID assignment.
- 261 • For hot-plug endpoints that have already received an EID assignment: After exiting any
262 temporary state where the hot-plug endpoint was unable to respond to MCTP control requests
263 for more than $T_{RECLAIM}$ seconds.

264 Only endpoints that have their Discovered flag set to *undiscovered* will respond to the Endpoint Discovery
265 message. Endpoints that have the flag set to *discovered* will not respond.

266 For PCIe endpoints, an Endpoint Discovery broadcast request message can be sent by the PCIe bus
267 owner to discover all MCTP-capable devices. MCTP-capable endpoints respond with an Endpoint
268 Discovery response message.

269 6.9.2 PCIe Endpoint Announcement

270 One or more endpoints may announce their presence and their need for an EID assignment by
271 autonomously sending a Discovery Notify message to the bus owner. This would typically trigger the bus
272 owner to perform the PCIe endpoint discovery/enumeration processes described in the following
273 subclauses.

274 **6.9.3 Full Endpoint Discovery/Enumeration**

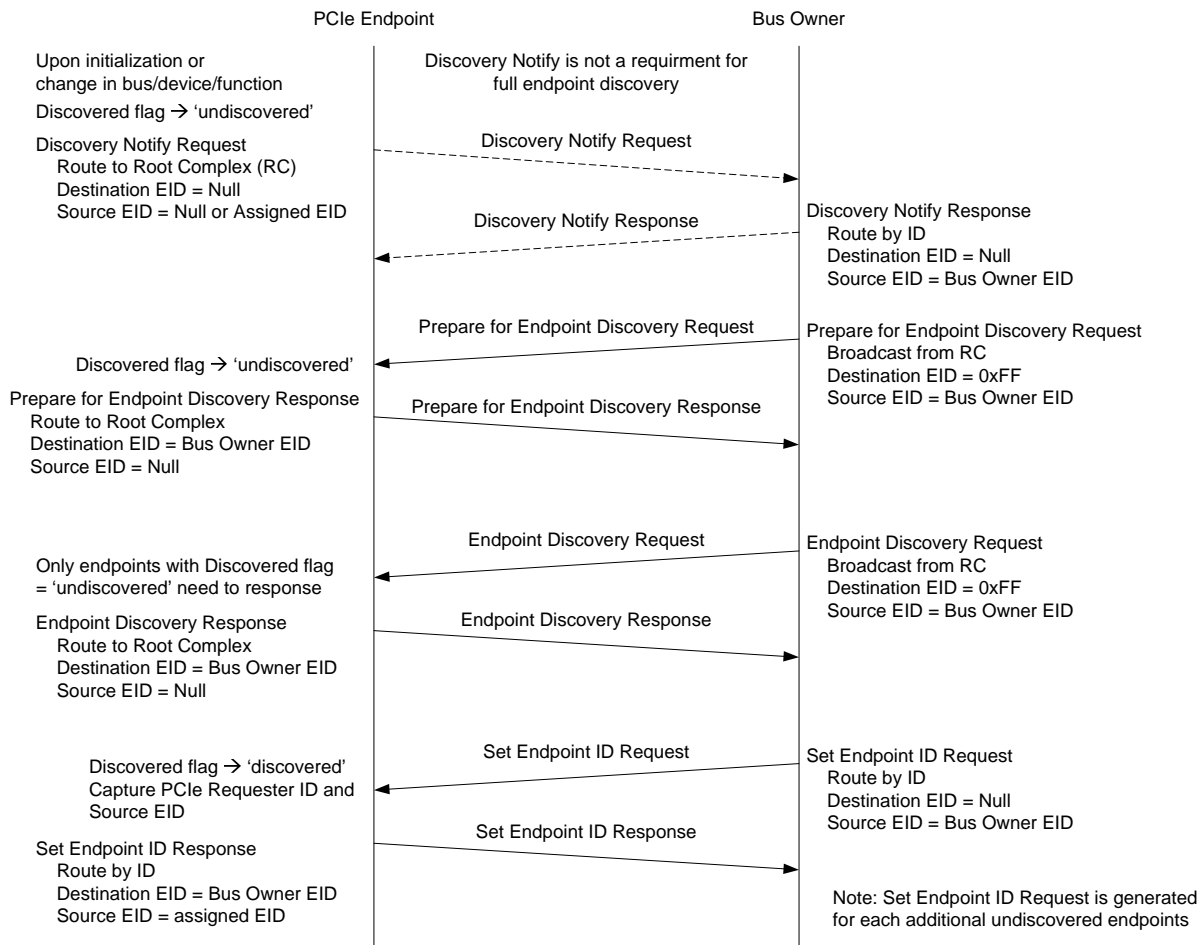
275 The following process is typically used when the bus owner wishes to discover and enumerate all
276 endpoints on the PCIe bus.

- 277 1) The PCIe bus owner issues a broadcast Prepare for Endpoint Discovery message. This
278 message causes each discoverable endpoint on the bus to set its PCIe endpoint Discovered
279 flag to undiscovered. Depending on the number of endpoints and the buffer space available in
280 the PCIe bus owner, the bus owner may not receive all of the response messages. The
281 discovery process does not require the bus owner to receive all the response messages to the
282 Prepare for Endpoint Discovery request. Because the PCIe bus owner can not determine that
283 all endpoints have received the Prepare for Endpoint Discovery request, it is recommended that
284 Prepare for Endpoint Discovery request is retried MN1 times to help ensure that all endpoints
285 have received the request. The PCIe bus owner is not required to wait for MT2 time interval
286 between the retries.
- 287 2) The PCIe bus owner should wait for MT2 time interval to help ensure that all endpoints that
288 received the Prepare for Endpoint Discovery request have processed the request.
- 289 3) The PCIe bus owner issues a broadcast Endpoint Discovery request message. All MCTP-
290 capable devices that have their Discovered flag set to undiscovered will respond with an
291 Endpoint Discovery response message.
- 292 4) Depending on the number of endpoints and the buffer space available in the bus owner, the bus
293 owner receives some or all of these response messages. For each response message received
294 from an undiscovered MCTP-capable device PCIe bus/device/function number, the bus owner
295 issues a Set Endpoint ID command to the physical address for the endpoint. This causes the
296 endpoint to set its Discovered flag to *discovered*. From this point, the endpoint will not respond
297 to the Endpoint Discovery command until another Prepare for Endpoint Discovery command is
298 received or some other condition causes the Discovered flag to be set back to *undiscovered*.
- 299 5) If The PCIe bus owner received any responses to the Endpoint Discovery request issued in
300 Step 3, then it repeats steps 3 and 4 until it no longer gets any responses to the Endpoint
301 Discovery request. In this case, then the PCIe bus owner is allowed to send the next Endpoint
302 Discovery request without waiting for MT2 time interval. If no responses were received by the
303 PCIe bus owner to the Endpoint Discovery request within the MT2 time interval, then the
304 discovery process is completed.

305 After the initial endpoint enumeration, it is recommended that the bus owner maintains a list of the unique
306 IDs for the endpoints it has discovered, and reassigns the same IDs to those endpoints if a
307 bus/device/function number changes during system operation.

308 Figure 2 provides an example flow of operations for full endpoint discovery.

Full PCIe MCTP Endpoint Discovery



309

310

Figure 2 – Flow of Operations for Full PCIe MCTP Discovery

311 **6.9.4 Partial Endpoint Discovery/Enumeration**

312 This process is used when the bus owner wishes to discover endpoints that may have been added to the
 313 bus after a full enumeration has been done. This situation can occur if a device has its
 314 bus/device/function number change after the full enumeration has been done, or when a hot-plug device
 315 is added to the system, or if a device that is already present in the system — but was in a disabled or
 316 powered-down state — comes on-line.

317 The partial discovery process is the same as the full discovery process except that the bus owner skips
 318 the step of broadcasting a Prepare for Endpoint Discovery command in order to avoid clearing the
 319 Discovered flags of already discovered endpoints.

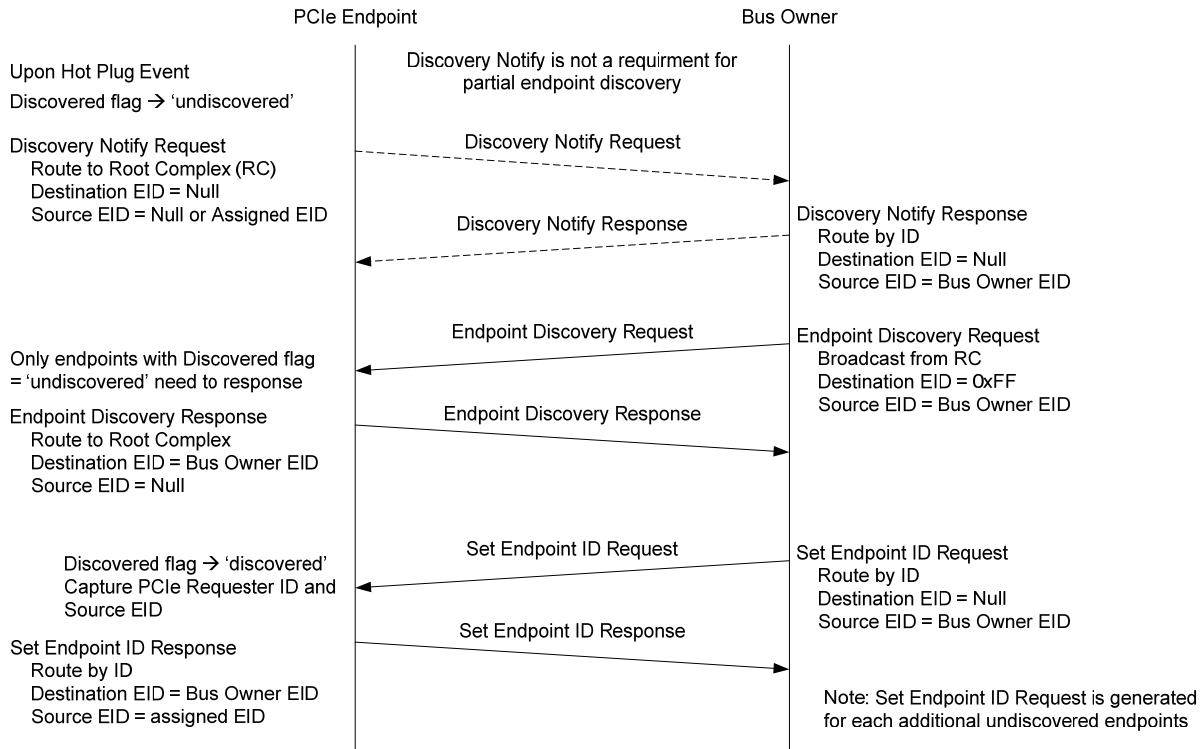
320 The partial discovery process may be initiated when a device that is added or enabled for MCTP sends a
 321 Discovery Notify message to the bus owner. The bus owner may also elect to periodically issue a
 322 broadcast Endpoint Discovery message to test for whether any undiscovered endpoints have been
 323 missed. The Discovery Notify message provides the bus owner with the bus/device/function number of
 324 the MCTP PCIe endpoint. The bus owner can then send a directed Endpoint Discovery message to the
 325 endpoint to confirm that the device has not been discovered. The bus owner then issues a Set Endpoint

326 ID command to the physical address for the endpoint which causes the endpoint to set its Discovered flag
 327 to *discovered*.

328 It is recommended that the bus owner maintains a list of the unique IDs for the endpoints it has
 329 discovered, and reassigns the same IDs to those endpoints if a bus/device/function number changes
 330 during system operation.

331 Figure 3 provides an example flow of operations for partial endpoint discovery.

Partial PCIe MCTP Endpoint Discovery



332

333

Figure 3 – Flow of Operations for Partial Endpoint Discovery

334 **6.9.5 Endpoint Re-enumeration**

335 If the bus implementation includes hot-plug devices, the bus owner shall perform a full or partial endpoint
 336 discovery any time the bus owner goes into a temporary state where the bus owner can miss receiving a
 337 Discovery Notify message (for example, if the bus owner device is reset or receives a firmware update).
 338 Whether a full or partial endpoint discovery is required is dependent on how much information the bus
 339 owner retains from prior enumerations.

340 **6.10 MCTP Messages Timing Requirements**

341 Table 4 lists MCTP-specific timing requirements for MCTP Control messages and operation on the PCIe
 342 VDM medium.

343 **Table 4 – Timing Specifications for MCTP Control Messages on PCIe VDM**

Timing Specification	Symbol	Min	Max	Description
Endpoint ID reclaim	TRECLAIM	–	5 sec	Maximum interval that a hot-plug endpoint is allowed to be non-responsive to MCTP control messages before its EID can be reclaimed by the bus owner.
Number of request retries	MN1	2	See Description column	Total of three tries, minimum: the original try plus two retries. The maximum number of retries for a given request is limited by the requirement that all retries shall occur within MT4, max of the initial request.
Request-to-response time	MT1	–	120 ms	This interval is measured at the responder from the end of the reception of an MCTP control request to the beginning of the transmission of the corresponding MCTP control response. This requirement is tested under the condition where the responder can successfully transmit the response on the first try.
Time-out waiting for a response	MT2	MT1 max ^[1] + 6 ms	MT4, min ^[1]	This interval at the requester sets the minimum amount of time that a requester should wait before retrying a MCTP control request. This interval is measured at the requester from the end of the successful transmission of the MCTP control request to the beginning of the reception of the corresponding MCTP control response. NOTE: This specification does not preclude an implementation from adjusting the minimum time-out waiting for a response to a smaller number than MT2 based on the measured response times from responders. The mechanism for doing so is outside the scope of this specification.
Instance ID expiration interval	MT4	5 sec ^[2]	6 sec	Interval after which the instance ID for a given response will expire and become reusable if a response has not been received for the request. This is also the maximum time that a responder tracks an instance ID for a given request from a given requester.
NOTE 1: Unless otherwise specified, this timing applies to the mandatory and optional MCTP commands.				
NOTE 2: If a requester is reset, it may produce the same sequence number for a request as one that was previously issued. To guard against this, it is recommended that sequence number expiration be implemented. Any request from a given requester that is received more than MT4 seconds after a previous, matching request should be treated as a new request, not a retry.				

Annex A (informative)

Notations and Conventions

344
345
346
347

348 A.1 Notations

349 Examples of notations used in this document are as follows: list into text needed

- 350 • 2:N In field descriptions, this will typically be used to represent a range of byte offsets
351 starting from byte two and continuing to and including byte N. The lowest offset is on
352 the left, the highest is on the right.
- 353 • (6) Parentheses around a single number can be used in message field descriptions to
354 indicate a byte field that may be present or absent.
- 355 • (3:6) Parentheses around a field consisting of a range of bytes indicates the entire range
356 may be present or absent. The lowest offset is on the left, the highest is on the right.
- 357 • PCIe Underlined, blue text is typically used to indicate a reference to a document or
358 specification called out in Clause 2, "Normative References" or to items hyperlinked
359 within the document.
- 360 • rsvd Abbreviation for Reserved. Case insensitive.
- 361 • [4] Square brackets around a number are typically used to indicate a bit offset. Bit offsets
362 are given as 0-based values (that is, the least significant bit [LSb] offset = 0).
- 363 • [7:5] A range of bit offsets. The most significant bit is on the left, the least significant bit is
364 on the right.
- 365 • 1b The lower case "b" following a number consisting of 0s and 1s is used to indicate the
366 number is being given in binary format.
- 367 • 0x12A A leading "0x" is used to indicate a number given in hexadecimal format.

368
369
370
371
372

Annex B (informative)

Change Log

Version	Date	Author	Description
1.0.0	7/28/2009		DMTF Standard Release

373