

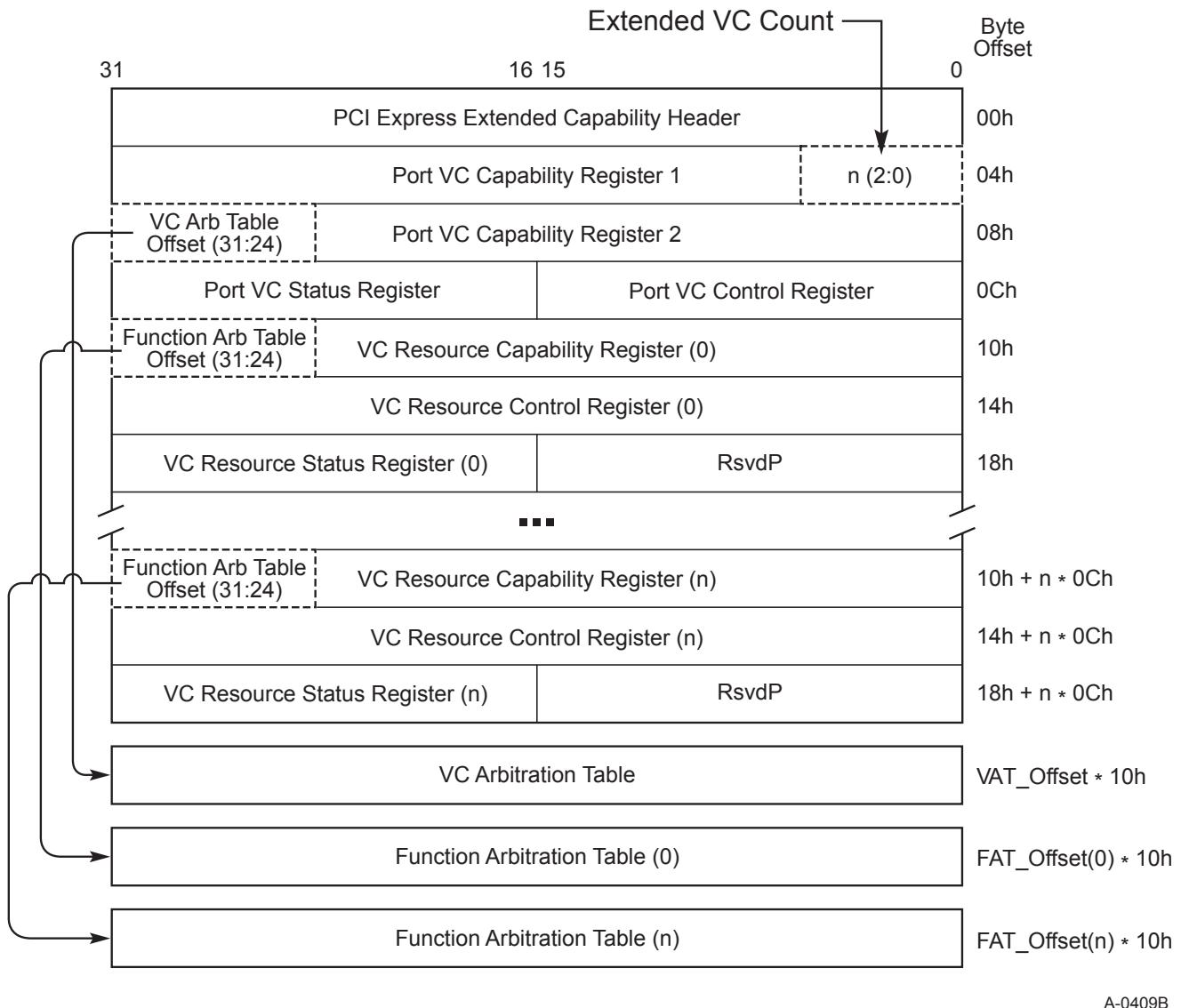
<u>Port Arbitration Select</u>	<u>Port Arbitration Table Length</u>
100b	128 entries
101b	256 entries

## 7.9.2 Multi-Function Virtual Channel Extended Capability

The Multi-Function Virtual Channel Extended Capability (**MFVC Capability**) is an optional Extended Capability that permits enhanced QoS management in a Multi-Function Device, including TC/VC mapping, optional VC arbitration, and optional Function arbitration for Upstream Requests. When implemented, the MFVC Capability structure must be present in the Extended Configuration Space of Function 0 of the Multi-Function Device's Upstream Port. Figure 7-182 provides a high level view of the MFVC Capability structure. This MFVC Capability structure controls Virtual Channel assignment at the PCI Express Upstream Port of the Multi-Function Device, while a VC Capability structure, if present in a Function, controls the Virtual Channel assignment for that individual Function.

The number of (extended) Virtual Channels is indicated by the Extended VC Count field in the Port VC Capability Register 1. Software must interpret this field to determine the availability of extended VC Resource registers.

A Multi-Function Device is permitted to have an MFVC Capability structure even if none of its Functions have a VC Capability structure. However, an MFVC Capability structure is permitted only in Function 0 in the Upstream Port of a Multi-Function Device.

Figure 7-182 MFVC Capability Structure

The following sections describe the registers/fields of the MFVC Capability structure.

### 7.9.2.1 MFVC Extended Capability Header (Offset 00h)

Refer to Section 7.6.3 for a description of the PCI Express Extended Capability header. The Extended Capability ID for the MFVC Capability is 0008h. Figure 7-183 details allocation of register fields in the MFVC Extended Capability header; Table 7-149 provides the respective bit definitions.

A-0409B

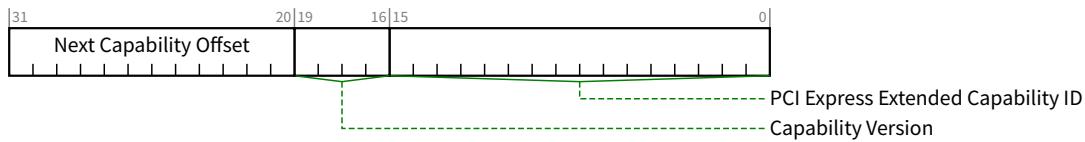


Figure 7-183 MFVC Extended Capability Header

Table 7-149 MFVC Extended Capability Header

Bit Location	Register Description	Attributes
15:0	<b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability.  The Extended Capability ID for the MFVC Capability is 0008h.	<u>RO</u>
19:16	<b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the Capability structure present.  Must be 1h for this version of the specification.	<u>RO</u>
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Capability structure or 000h if no other items exist in the linked list of Capabilities.  For Extended Capabilities implemented in Configuration Space, this offset is relative to the beginning of PCI-compatible Configuration Space and thus must always be either 000h (for terminating list of Capabilities) or greater than OFFh.	<u>RO</u>

### 7.9.2.2 MFVC Port VC Capability Register 1 (Offset 04h)

The MFVC Port VC Capability Register 1 describes the configuration of the Virtual Channels associated with a PCI Express Port of the Multi-Function Device. Figure 7-184 details allocation of register fields in the MFVC Port VC Capability Register 1; Table 7-150 provides the respective bit definitions.

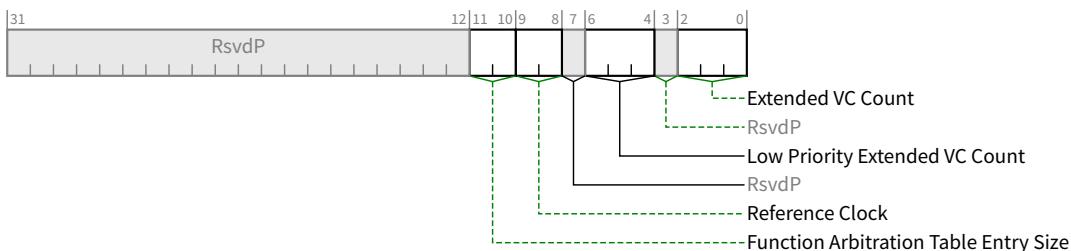


Figure 7-184 MFVC Port VC Capability Register 1

*Table 7-150 MFVC Port VC Capability Register 1*

Bit Location	Register Description	Attributes
2:0	<p><b>Extended VC Count</b> - Indicates the number of (extended) Virtual Channels in addition to the default VC supported by the device.</p> <p>This value indicates the number of (extended) VC Resource Capability, Control, and Status registers that are present in Configuration Space in addition to the required VC Resource registers for the default VC.</p> <p>The minimum value of this field is 0 (for devices that only support the default VC and only have 1 set of VC Resource Registers for that VC). The maximum value is 7.</p>	<u>RO</u>
6:4	<p><b>Low Priority Extended VC Count</b> - Indicates the number of (extended) Virtual Channels in addition to the default VC belonging to the low-priority VC (LPVC) group that has the lowest priority with respect to other VC resources in a strict-priority VC Arbitration.</p> <p>The minimum value of this field is 000b and the maximum value is <u>Extended VC Count</u>.</p>	<u>RO</u>
9:8	<p><b>Reference Clock</b> - Indicates the reference clock for Virtual Channels that support time-based WRR Function Arbitration.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>00b</b> 100 ns reference clock</li> <li><b>01b - 11b</b> Reserved</li> </ul>	<u>RO</u>
11:10	<p><b>Function Arbitration Table Entry Size</b> - Indicates the size (in bits) of Function Arbitration table entry in the device.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>00b</b> Size of Function Arbitration table entry is 1 bit</li> <li><b>01b</b> Size of Function Arbitration table entry is 2 bits</li> <li><b>10b</b> Size of Function Arbitration table entry is 4 bits</li> <li><b>11b</b> Size of Function Arbitration table entry is 8 bits</li> </ul>	<u>RO</u>

### 7.9.2.3 MFVC Port VC Capability Register 2 (Offset 08h)

The MFVC Port VC Capability Register 2 provides further information about the configuration of the Virtual Channels associated with a PCI Express Port of the Multi-Function Device. Figure 7-185 details allocation of register fields in the MFVC Port VC Capability Register 2; Table 7-151 provides the respective bit definitions.

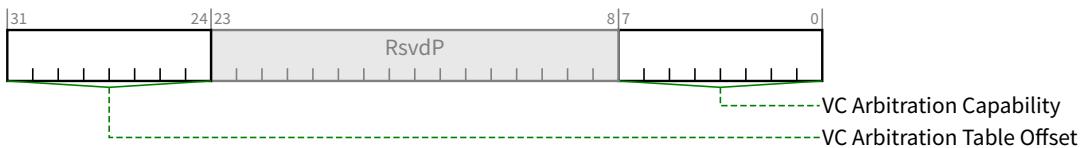
*Figure 7-185 MFVC Port VC Capability Register 2*

Table 7-151 MFVC Port VC Capability Register 2

Bit Location	Register Description	Attributes
7:0	<p><b>VC Arbitration Capability</b> - Indicates the types of VC Arbitration supported by the device for the LPVC group. This field is valid for all devices that report a <u>Low Priority Extended VC Count</u> greater than 0.</p> <p>Each Bit Location within this field corresponds to a <u>VC Arbitration Capability</u> defined below. When more than 1 bit in this field is Set, it indicates that the device can be configured to provide different VC arbitration services.</p> <p>Defined bit positions are:</p> <ul style="list-style-type: none"> <li><b>Bit 0</b> Hardware fixed arbitration scheme, e.g., Round Robin</li> <li><b>Bit 1</b> Weighted Round Robin (WRR) arbitration with 32 phases</li> <li><b>Bit 2</b> WRR arbitration with 64 phases</li> <li><b>Bit 3</b> WRR arbitration with 128 phases</li> <li><b>Bits 4-7</b> Reserved</li> </ul>	RO
31:24	<p><b>VC Arbitration Table Offset</b> - Indicates the location of the <u>MFVC VC Arbitration Table</u>.</p> <p>This field contains the zero-based offset of the table in DQWORDS (16 bytes) from the base address of the MFVC Capability structure. A value of 00h indicates that the table is not present.</p>	RO

#### 7.9.2.4 MFVC Port VC Control Register (Offset 0Ch)

Figure 7-186 details allocation of register fields in the Port VC Control register; Table 7-152 provides the respective bit definitions.

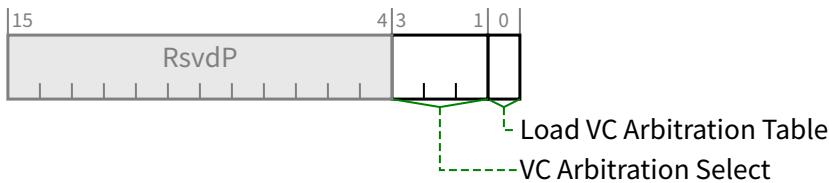


Figure 7-186 MFVC Port VC Control Register

Table 7-152 MFVC Port VC Control Register

Bit Location	Register Description	Attributes
0	<p><b>Load VC Arbitration Table</b> - Used by software to update the <u>MFVC VC Arbitration Table</u>. This bit is valid when the selected VC Arbitration uses the <u>MFVC VC Arbitration Table</u>.</p> <p>Software Sets this bit to request hardware to apply new values programmed into <u>MFVC VC Arbitration Table</u>; Clearing this bit has no effect. Software checks the <u>VC Arbitration Table Status</u> bit to confirm that new values stored in the <u>MFVC VC Arbitration Table</u> are latched by the VC arbitration logic.</p> <p>This bit always returns 0b when read.</p>	RW
3:1	<p><b>VC Arbitration Select</b> - Used by software to configure the VC arbitration by selecting one of the supported VC Arbitration schemes indicated by the <u>VC Arbitration Capability</u> field in the <u>MFVC Port VC Capability Register 2</u>.</p>	RW

Bit Location	Register Description	Attributes
	<p>The permissible values of this field are numbers corresponding to one of the asserted bits in the <u>VC Arbitration Capability</u> field.</p> <p>This field cannot be modified when more than one VC in the LPVC group is enabled.</p>	

### 7.9.2.5 MFVC Port VC Status Register (Offset 0Eh)

The MFVC Port VC Status Register provides status of the configuration of Virtual Channels associated with a Port of the Multi-Function Device. Figure 7-187 details allocation of register fields in the MFVC Port VC Status Register; Table 7-153 provides the respective bit definitions.

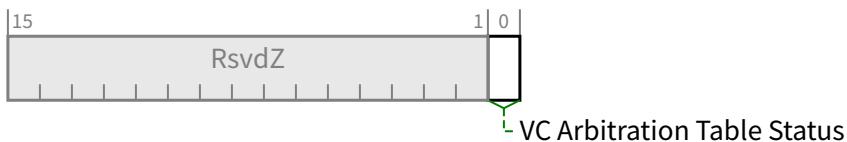


Figure 7-187 MFVC Port VC Status Register

Table 7-153 MFVC Port VC Status Register

Bit Location	Register Description	Attributes
0	<p><b>VC Arbitration Table Status</b> - Indicates the coherency status of the MFVC VC Arbitration Table. This bit is valid when the selected VC uses the MFVC VC Arbitration Table.</p> <p>This bit is Set by hardware when any entry of the MFVC VC Arbitration Table is written by software. This bit is Cleared by hardware when hardware finishes loading values stored in the MFVC VC Arbitration Table after software sets the Load VC Arbitration Table bit in the MFVC Port VC Control Register.</p> <p>Default value of this bit is 0b.</p>	RO

### 7.9.2.6 MFVC VC Resource Capability Register

The MFVC VC Resource Capability Register describes the capabilities and configuration of a particular Virtual Channel resource. Figure 7-188 details allocation of register fields in the MFVC VC Resource Capability Register; Table 7-154 provides the respective bit definitions.

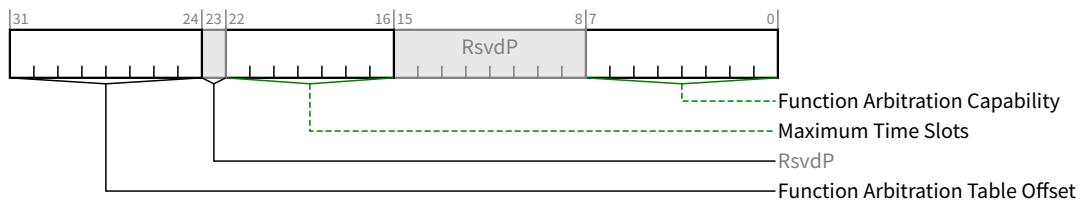


Figure 7-188 MFVC VC Resource Capability Register

Table 7-154 MFVC VC Resource Capability Register

Bit Location	Register Description	Attributes
7:0	<p><b>Function Arbitration Capability</b> - Indicates types of Function Arbitration supported by the VC resource. Each Bit Location within this field corresponds to a <u>Function Arbitration Capability</u> defined below. When more than 1 bit in this field is Set, it indicates that the VC resource can be configured to provide different arbitration services.</p> <p>Software selects among these capabilities by writing to the <u>Function Arbitration Select</u> field (see <a href="#">Section 7.9.2.7</a>).</p> <p>Defined bit positions are:</p> <ul style="list-style-type: none"> <li><b>Bit 0</b> Non-configurable hardware-fixed arbitration scheme, e.g., Round Robin (RR)</li> <li><b>Bit 1</b> Weighted Round Robin (WRR) arbitration with 32 phases</li> <li><b>Bit 2</b> WRR arbitration with 64 phases</li> <li><b>Bit 3</b> WRR arbitration with 128 phases</li> <li><b>Bit 4</b> Time-based WRR with 128 phases</li> <li><b>Bit 5</b> WRR arbitration with 256 phases</li> <li><b>Bits 6-7</b> Reserved</li> </ul>	RO
22:16	<p><b>Maximum Time Slots</b> - Indicates the maximum number of time slots (minus 1) that the VC resource is capable of supporting when it is configured for time-based WRR Function Arbitration. For example, a value of 000 0000b in this field indicates the supported maximum number of time slots is 1 and a value of 111 1111b indicates the supported maximum number of time slots is 128.</p> <p>This field is valid only when the <u>Function Arbitration Capability</u> indicates that the VC resource supports time-based WRR Function Arbitration.</p>	HwInit
31:24	<p><b>Function Arbitration Table Offset</b> - Indicates the location of the Function Arbitration Table associated with the VC resource.</p> <p>This field contains the zero-based offset of the table in DQWORDS (16 bytes) from the base address of the MFVC Capability structure. A value of 00h indicates that the table is not present.</p>	RO

### 7.9.2.7 MFVC VC Resource Control Register

Figure 7-189 details allocation of register fields in the MFVC VC Resource Control Register; Table 7-155 provides the respective bit definitions.

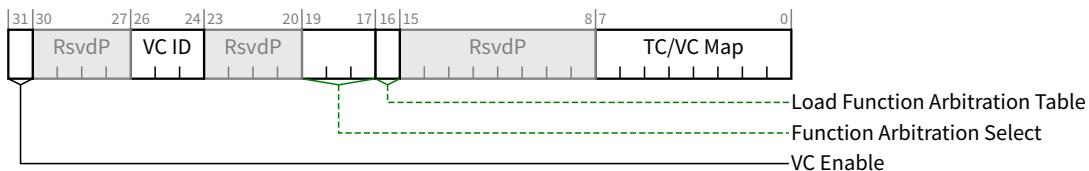


Figure 7-189 MFVC VC Resource Control Register

Table 7-155 MFVC VC Resource Control Register

Bit Location	Register Description	Attributes
7:0	<p><b>TC/VC Map</b> - This field indicates the TCs that are mapped to the VC resource. Bit Locations within this field correspond to TC values. For example, when bit 7 is Set in this field, TC7 is mapped to this VC resource. When more than 1 bit in this field is Set, it indicates that multiple TCs are mapped to the VC resource.</p> <p>In order to remove one or more TCs from the <u>TC/VC Map</u> of an enabled VC, software must ensure that no new or outstanding transactions with the TC labels are targeted at the given Link.</p> <p>Default value of this field is FFh for the first VC resource and is 00h for other VC resources.</p> <p><b>Note:</b></p> <p>Bit 0 of this field is read-only. It must be hardwired to 1b for the default VC0 and hardwired to 0b for all other enabled VCs.</p>	RW (see the note for exceptions)
16	<p><b>Load Function Arbitration Table</b> - When Set, this bit updates the Function Arbitration logic from the Function Arbitration Table for the VC resource. This bit is only valid when the Function Arbitration Table is used by the selected Function Arbitration scheme (that is indicated by a Set bit in the <u>Function Arbitration Capability</u> field selected by <u>Function Arbitration Select</u>).</p> <p>Software sets this bit to signal hardware to update Function Arbitration logic with new values stored in the Function Arbitration Table; clearing this bit has no effect. Software uses the Function Arbitration Table Status bit to confirm whether the new values of Function Arbitration Table are completely latched by the arbitration logic.</p> <p>This bit always returns 0b when read.</p> <p>Default value of this bit is 0b.</p>	RW
19:17	<p><b>Function Arbitration Select</b> - This field configures the VC resource to provide a particular Function Arbitration service.</p> <p>The permissible value of this field is a number corresponding to one of the asserted bits in the <u>Function Arbitration Capability</u> field of the VC resource.</p>	RW
26:24	<p><b>VC ID</b> - This field assigns a <u>VC ID</u> to the VC resource (see note for exceptions).</p> <p>This field cannot be modified when the VC is already enabled.</p> <p><b>Note:</b></p> <p>For the first VC resource (default VC), this field is a read-only field that must be hardwired to 000b.</p>	RW
31	<p><b>VC Enable</b> - When Set, this bit enables a Virtual Channel (see note 1 for exceptions). The Virtual Channel is disabled when this bit is cleared.</p> <p>Software must use the <u>VC Negotiation Pending</u> bit to check whether the VC negotiation is complete.</p> <p>Default value of this bit is 1b for the first VC resource and 0b for other VC resource(s).</p>	RW

Bit Location	Register Description	Attributes
	<p>Notes:</p> <ol style="list-style-type: none"> <li>1. This bit is hardwired to 1b for the default VC (VC0), i.e., writing to this field has no effect for VC0.</li> <li>2. To enable a Virtual Channel, the <u>VC Enable</u> bits for that Virtual Channel must be Set in both components on a Link.</li> <li>3. To disable a Virtual Channel, the <u>VC Enable</u> bits for that Virtual Channel must be Cleared in both components on a Link.</li> <li>4. Software must ensure that no traffic is using a Virtual Channel at the time it is disabled.</li> <li>5. Software must fully disable a Virtual Channel in both components on a Link before re-enabling the Virtual Channel.</li> </ol>	

### 7.9.2.8 MFVC VC Resource Status Register

Figure 7-190 details allocation of register fields in the MFVC VC Resource Status Register; Table 7-156 provides the respective bit definitions.

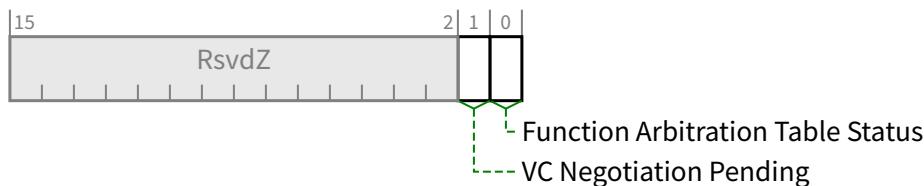


Figure 7-190 MFVC VC Resource Status Register

Table 7-156 MFVC VC Resource Status Register

Bit Location	Register Description	Attributes
0	<p><b>Function Arbitration Table Status</b> - This bit indicates the coherency status of the Function Arbitration Table associated with the VC resource. This bit is valid only when the Function Arbitration Table is used by the selected Function Arbitration for the VC resource.</p> <p>This bit is Set by hardware when any entry of the Function Arbitration Table is written to by software. This bit is Cleared by hardware when hardware finishes loading values stored in the Function Arbitration Table after software sets the <u>Load Function Arbitration Table</u> bit.</p> <p>Default value of this bit is 0b.</p>	RO
1	<p><b>VC Negotiation Pending</b> - This bit indicates whether the Virtual Channel negotiation (initialization or disabling) is in pending state.</p> <p>When this bit is Set by hardware, it indicates that the VC resource is still in the process of negotiation. This bit is Cleared by hardware after the VC negotiation is complete. For a non-default Virtual Channel, software may use this bit when enabling or disabling the VC. For the default VC, this bit indicates the status of the process of Flow Control initialization.</p>	RO

Bit Location	Register Description	Attributes
	Before using a Virtual Channel, software must check whether the <u>VC Negotiation Pending</u> bits for that Virtual Channel are Clear in both components on a Link.	

### 7.9.2.9 MFVC VC Arbitration Table

The definition of the MFVC VC Arbitration Table in the MFVC Capability structure is identical to that in the VC Capability structure (see Section 7.9.1.9).

### 7.9.2.10 Function Arbitration Table

The Function Arbitration Table register in the MFVC Capability structure takes the same form as the Port Arbitration Table register in the VC Capability structure (see Section 7.9.1.10).

The Function Arbitration Table register is a read-write register array that is used to store the WRR or time-based WRR arbitration table for Function Arbitration for the VC resource. It is only present when one or more asserted bits in the Function Arbitration Capability field indicate that the Multi-Function Device supports a Function Arbitration scheme that uses a programmable arbitration table. Furthermore, it is only valid when one of the above-mentioned bits in the Function Arbitration Capability field is selected by the Function Arbitration Select field.

The Function Arbitration Table represents one Function arbitration period. Each table entry containing a Function Number or Function Group<sup>155</sup> Number corresponds to a phase within a Function Arbitration period. The table entry size requirements are as follows:

- The table entry size for non-ARI devices must support enough values to specify all implemented Functions plus at least one value that does not correspond to an implemented Function. For example, a table with 2-bit entries can be used by a Multi-Function Device with up to three Functions.
- The table entry size for ARI Devices must be either 4 bits or 8 bits.
  - If MFVC Function Groups are enabled, each entry maps to a single Function Group. Arbitration between multiple Functions within a Function Group is implementation specific, but must guarantee forward progress.
  - If MFVC Function Groups are not enabled and 4-bit entries are implemented, a given entry maps to all Functions whose Function Number modulo 8 matches its value. Similarly, if 8-bit entries are implemented, a given entry maps to all Functions whose Function Number modulo 128 matches its value. If a given entry maps to multiple Functions, arbitration between those Functions is implementation specific, but must guarantee forward progress.

A Function Number or Function Group Number written to a table entry indicates that the phase within the Function Arbitration period is assigned to the selected Function or Function Group (the Function Number or Function Group Number must be a valid one).

- When the WRR Function Arbitration is used for a VC of the Egress Port of the Multi-Function Device, at each arbitration phase the Function Arbiter serves one transaction from the Function or Function Group indicated by the Function Number or Function Group Number of the current phase. When finished, it immediately advances to the next phase. A phase is skipped, i.e., the Function Arbiter simply moves to the next phase immediately if the Function or Function Group indicated by the phase does not contain any transaction for the VC.

155. If an ARI Device supports MFVC Function Groups capability and ARI-aware software enables it, arbitration is based on Function Groups instead of Functions. See Section 7.8.7.

- When the Time-based WRR Function Arbitration is used for a VC of the Egress Port of the Multi-Function Device, at each arbitration phase aligning to a virtual timeslot, the Function Arbiter serves one transaction from the Function or Function Group indicated by the Function Number or Function Group Number of the current phase. It advances to the next phase at the next virtual timeslot. A phase indicates an “idle” timeslot, i.e., the Function Arbiter does not serve any transaction during the phase, if:
  - the phase contains the Number of a Function or a Function Group that does not exist, or
  - the Function or Function Group indicated by the phase does not contain any transaction for the VC.

The Function Arbitration Table Entry Size field in the MFVC Port VC Capability Register 1 determines the table entry size. The length of the table is determined by the Function Arbitration Select field as shown in Table 7-157 .

When the Function Arbitration Table is used by the default Function Arbitration for the default VC, the default values for the table entries must contain at least one entry for each of the active Functions or Function Groups in the Multi-Function Device to ensure forward progress for the default VC for the Multi-Function Device’s Upstream Port. The table may contain RR or RR-like fair Function Arbitration for the default VC.

*Table 7-157 Length of Function Arbitration Table*

Function Arbitration Select	Function Arbitration Table Length
001b	32 entries
010b	64 entries
011b	128 entries
100b	128 entries
101b	256 entries

### 7.9.3 Device Serial Number Extended Capability

The Device Serial Number Extended Capability is an optional Extended Capability that may be implemented by any PCI Express device Function. The Device Serial Number is a read-only 64-bit value that is unique for a given PCI Express device. Figure 7-191 details allocation of register fields in the Device Serial Number Extended Capability structure.

It is permitted but not recommended for RCiEPs to implement this Capability.

RCiEPs that implement this Capability are permitted but not required to return the same Device Serial Number value as that reported by other RCiEPs of the same Root Complex.

All Multi-Function DeviceDevices other than RCiEPs that implement this Capability must implement it for Function 0; other Functions that implement this Capability must return the same Device Serial Number value as that reported by Function 0.

RCiEPs are permitted to implement or not implement this Capability on an individual basis, independent of whether they are part of a Multi-Function Device.

A PCI Express component other than a Root Complex containing multiple Devices such as a PCI Express Switch that implements this Capability must return the same Device Serial Number for each device.

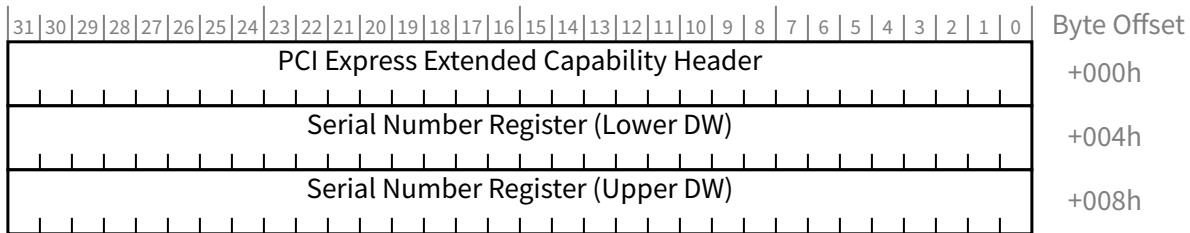


Figure 7-191 Device Serial Number Extended Capability Structure

### 7.9.3.1 Device Serial Number Extended Capability Header (Offset 00h)

Figure 7-192 details allocation of register fields in the Device Serial Number Extended Capability Header; Table 7-158 provides the respective bit definitions. Refer to Section 7.6.3 for a description of the PCI Express Extended Capability header. The Extended Capability ID for the Device Serial Number Extended Capability is 0003h.

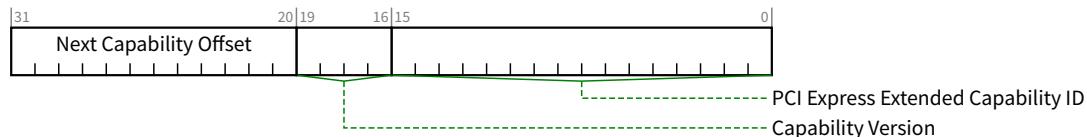


Figure 7-192 Device Serial Number Extended Capability Header

Table 7-158 Device Serial Number Extended Capability Header

Bit Location	Register Description	Attributes
15:0	<b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability.  Extended Capability ID for the Device Serial Number Extended Capability is 0003h.	RO
19:16	<b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the Capability structure present.  Must be 1h for this version of the specification.	RO
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Capability structure or 000h if no other items exist in the linked list of Capabilities.  For Extended Capabilities implemented in Configuration Space, this offset is relative to the beginning of PCI-compatible Configuration Space and thus must always be either 000h (for terminating list of Capabilities) or greater than OFFh.	RO

### 7.9.3.2 Serial Number Register (Offset 04h)

The Serial Number register is a 64-bit field that contains the IEEE defined 64-bit extended unique identifier [EUI-64]. Figure 7-193 details allocation of register fields in the Serial Number register; Table 7-159 provides the respective bit definitions.

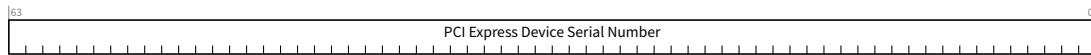


Figure 7-193 Serial Number Register

Table 7-159 Serial Number Register

Bit Location	Register Description	Attributes
63:0	<p><b>PCI Express Device Serial Number</b> - This field contains the IEEE defined 64-bit Extended Unique Identifier [EUI-64]. This identifier includes a 24-bit company id value assigned by IEEE registration authority and a 40-bit extension identifier assigned by the manufacturer.</p> <p>PCI Express Device Serial Number[07:00] = EUI[63:56]            PCI Express Device Serial Number[15:08] = EUI[55:48]            PCI Express Device Serial Number[23:16] = EUI[47:40]            PCI Express Device Serial Number[31:24] = EUI[39:32]            PCI Express Device Serial Number[39:32] = EUI[31:24]            PCI Express Device Serial Number[47:40] = EUI[23:16]            PCI Express Device Serial Number[55:48] = EUI[15:08]            PCI Express Device Serial Number[63:56] = EUI[07:00]</p>	RO

### 7.9.4 Vendor-Specific Capability

The Vendor-Specific Capability is a capability structure in PCI-compatible Configuration Space (first 256 bytes) as shown in Figure 7-194.

The Vendor-Specific Capability allows device vendors to use the Capability mechanism for vendor-specific information. The layout of the information is vendor-specific, except for the first three bytes, as explained below.

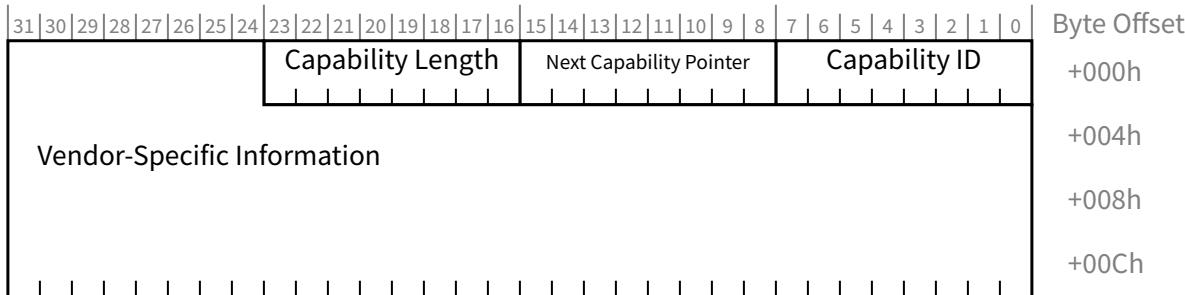


Figure 7-194 Vendor-Specific Capability

*Table 7-160 Vendor-Specific Capability*

Bit Location	Register Description	Attributes
7:0	<b>Capability ID</b> - Indicates the PCI Express Capability structure. This field must return a Capability ID of 09h indicating that this is a <u>Vendor-Specific Capability</u> structure.	<u>RO</u>
15:8	<b>Next Capability Pointer</b> - This field contains the offset to the next PCI Capability structure or 00h if no other items exist in the linked list of Capabilities.	<u>RO</u>
23:16	<b>Capability Length</b> - This field provides the number of bytes in the Capability structure (including the three bytes consumed by the Capability ID, Next Capability Pointer, and Capability Length field).	<u>RO</u>
31:24	<b>Vendor Specific Information</b>	Vendor Specific

## 7.9.5 Vendor-Specific Extended Capability

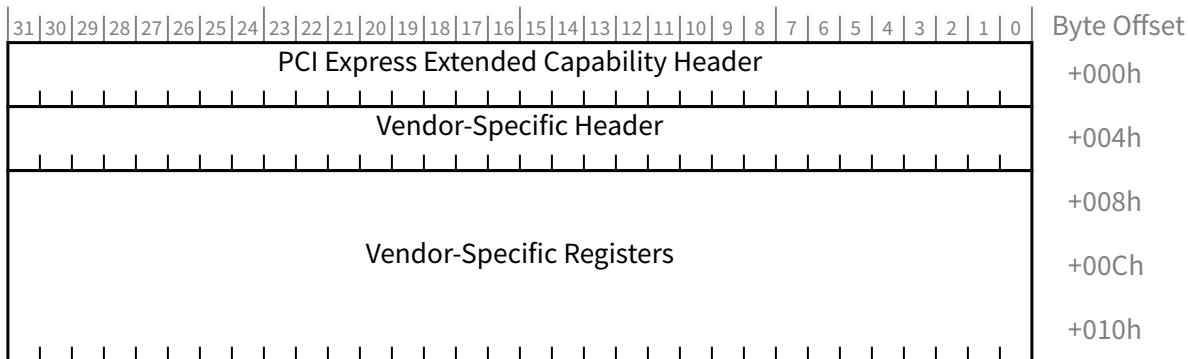
The Vendor-Specific Extended Capability (**VSEC Capability**) is an optional Extended Capability that is permitted to be implemented by any PCI Express Function or RCRB. This allows PCI Express component vendors to use the Extended Capability mechanism to expose vendor-specific registers.

A single PCI Express Function or RCRB is permitted to contain multiple VSEC structures.

An example usage is a set of vendor-specific features that are intended to go into an on-going series of components from that vendor. A VSEC structure can tell vendor-specific software which features a particular component supports, including components developed after the software was released.

Figure 7-195 details allocation of register fields in the VSEC structure. The structure of the Vendor-Specific Extended Capability Header and the Vendor-Specific Header is architected by this specification.

With a PCI Express Function, the structure and definition of the vendor-specific Registers area is determined by the vendor indicated by the Vendor ID field located at byte offset 00h in PCI-compatible Configuration Space. With an RCRB, a VSEC is permitted only if the RCRB also contains an RCRB Header Extended Capability structure, which contains a Vendor ID field indicating the vendor.

*Figure 7-195 VSEC Capability Structure*

### 7.9.5.1 Vendor-Specific Extended Capability Header (Offset 00h)

Figure 7-196 details allocation of register fields in the Vendor-Specific Extended Capability Header; Table 7-161 provides the respective bit definitions. Refer to Section 7.6.3 for a description of the PCI Express Extended Capability Header. The Extended Capability ID for the Vendor-Specific Extended Capability is 000Bh.

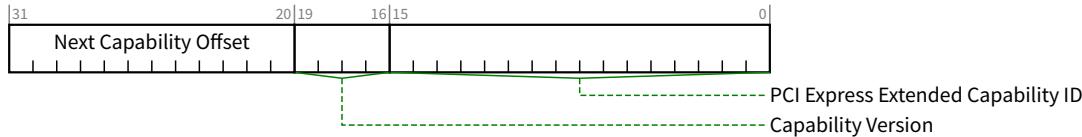


Figure 7-196 Vendor-Specific Extended Capability Header

Table 7-161 Vendor-Specific Extended Capability Header

Bit Location	Register Description	Attributes
15:0	<b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability. Extended Capability ID for the Vendor-Specific Extended Capability is 000Bh.	RO
19:16	<b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the Capability structure present. Must be 1h for this version of the specification.	RO
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Capability structure or 000h if no other items exist in the linked list of Capabilities. For Extended Capabilities implemented in Configuration Space, this offset is relative to the beginning of PCI-compatible Configuration Space and thus must always be either 000h (for terminating list of Capabilities) or greater than OFFh.	RO

### 7.9.5.2 Vendor-Specific Header (Offset 04h)

Figure 7-197 details allocation of register fields in the Vendor-Specific Header; Table 7-162 provides the respective bit definitions.

Vendor-specific software must qualify the associated Vendor ID of the PCI Express Function or RCRB before attempting to interpret the values in the VSEC ID or VSEC Rev fields.

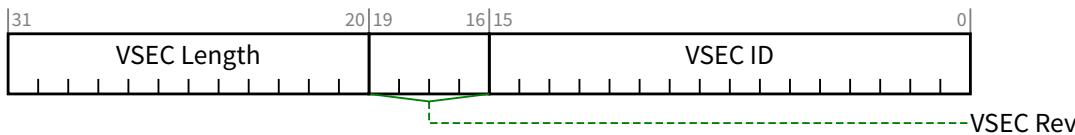


Figure 7-197 Vendor-Specific Header

*Table 7-162 Vendor-Specific Header*

Bit Location	Register Description	Attributes
15:0	<b>VSEC ID</b> - This field is a vendor-defined ID number that indicates the nature and format of the VSEC structure.  Software must qualify the <u>Vendor ID</u> before interpreting this field.	<u>RO</u>
19:16	<b>VSEC Rev</b> - This field is a vendor-defined version number that indicates the version of the VSEC structure.  Software must qualify the <u>Vendor ID</u> and <u>VSEC ID</u> before interpreting this field.	<u>RO</u>
31:20	<b>VSEC Length</b> - This field indicates the number of bytes in the entire VSEC structure, including the <u>Vendor-Specific Extended Capability Header</u> , the <u>Vendor-Specific Header</u> , and the vendor-specific registers.	<u>RO</u>

## 7.9.6 Designated Vendor-Specific Extended Capability (DVSEC)

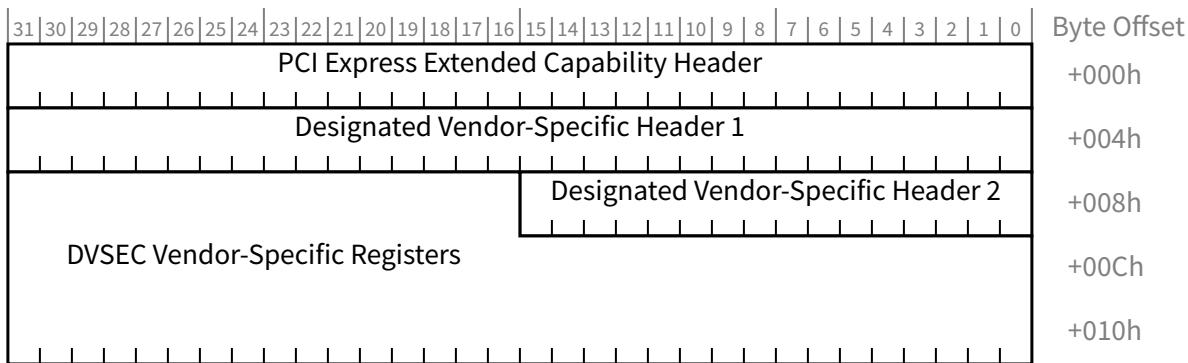
The Designated Vendor-Specific Extended Capability (**DVSEC Capability**) is an optional Extended Capability that is permitted to be implemented by any PCI Express Function or RCRB. This allows PCI Express component vendors to use the Extended Capability mechanism to expose vendor-specific registers that can be present in components by a variety of vendors.

A single PCI Express Function or RCRB is permitted to contain multiple DVSEC Capability structures.

An example usage is a set of vendor-specific features that are intended to go into an on-going series of components from a collection of vendors. A DVSEC Capability structure can tell vendor-specific software which features a particular component supports, including components developed after the software was released.

Figure 7-198 details allocation of register fields in the DVSEC Capability structure. The structure of the PCI Express Extended Capability Header and the Designated Vendor-Specific header is architected by this specification.

The DVSEC Vendor-Specific Register area begins at offset 0Ah.

*Figure 7-198 Designated Vendor-Specific Extended Capability*

### 7.9.6.1 Designated Vendor-Specific Extended Capability Header (Offset 00h)

Figure 7-199 details allocation of register fields in the Designated Vendor-Specific Extended Capability Header; Table 7-163 provides the respective bit definitions. Refer to Section 7.9.3 for a description of the PCI Express Extended Capability Header. The Extended Capability ID for the Designated Vendor-Specific Extended Capability is 0023h.

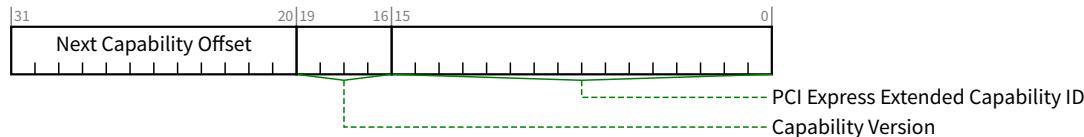


Figure 7-199 Designated Vendor-Specific Extended Capability Header

Table 7-163 Designated Vendor-Specific Extended Capability Header

Bit Location	Register Description	Attributes
15:0	<b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability. Extended Capability ID for the Designated Vendor-Specific Extended Capability is 0023h.	RO
19:16	<b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the Capability structure present. Must be 1h for this version of the specification.	RO
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Capability structure or 000h if no other items exist in the linked list of Capabilities. For Extended Capabilities implemented in Configuration Space, this offset is relative to the beginning of PCI-compatible Configuration Space and thus must always be either 000h (for terminating list of Capabilities) or greater than OFFh.	RO

### 7.9.6.2 Designated Vendor-Specific Header 1 (Offset 04h)

Figure 7-200 details allocation of register fields in the Designated Vendor-Specific Header 1; Table 7-164 provides the respective bit definitions.

Vendor-specific software must qualify the DVSEC Vendor ID before attempting to interpret the DVSEC Revision field.

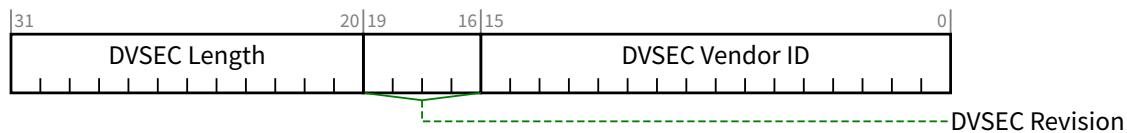


Figure 7-200 Designated Vendor-Specific Header 1

*Table 7-164 Designated Vendor-Specific Header 1*

Bit Location	Register Description	Attributes
15:0	<b>DVSEC Vendor ID</b> - This field is the Vendor ID associated with the vendor that defined the contents of this capability.	RO
19:16	<b>DVSEC Revision</b> - This field is a vendor-defined version number that indicates the version of the DVSEC structure. Software must qualify the <u>DVSEC Vendor ID</u> and <u>DVSEC ID</u> before interpreting this field.	RO
31:20	<b>DVSEC Length</b> - This field indicates the number of bytes in the entire DVSEC structure, including the PCI Express Extended Capability Header, the DVSEC Header 1, DVSEC Header 2, and DVSEC vendor-specific registers.	RO

### 7.9.6.3 Designated Vendor-Specific Header 2 (Offset 08h)

Figure 7-201 details allocation of register fields in the Designated Vendor-Specific Header 2; Table 7-165 provides the respective bit definitions.

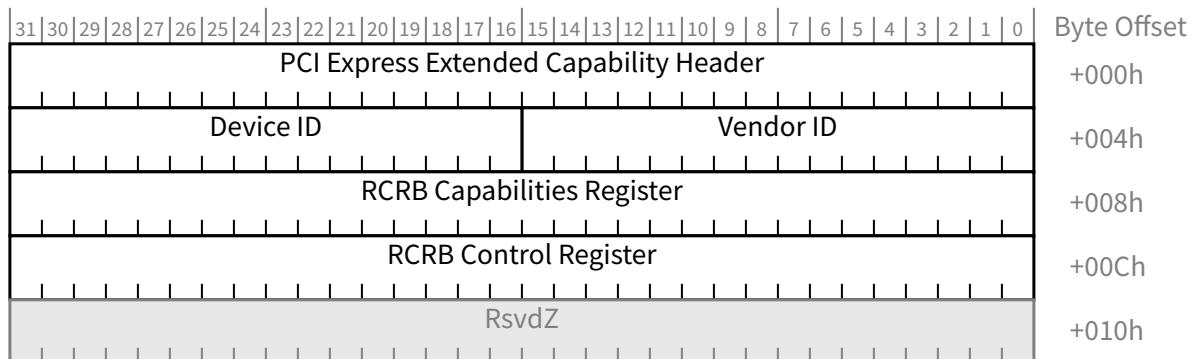
Vendor-specific software must qualify the DVSEC Vendor ID before attempting to interpret the DVSEC ID field.

*Figure 7-201 Designated Vendor-Specific Header 2**Table 7-165 Designated Vendor-Specific Header 2*

Bit Location	Register Description	Attributes
15:0	<b>DVSEC ID</b> - This field is a vendor-defined ID that indicates the nature and format of the DVSEC structure. Software must qualify the <u>DVSEC Vendor ID</u> before interpreting this field.	RO

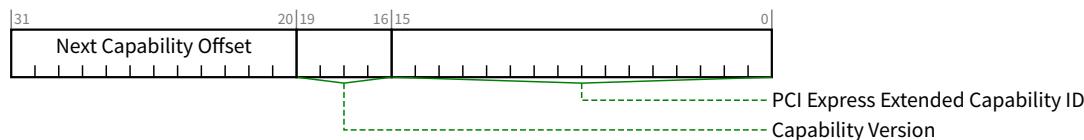
### 7.9.7 RCRB Header Extended Capability

The PCI Express RCRB Header Extended Capability is an optional Extended Capability that may be implemented in an RCRB to provide a Vendor ID and Device ID for the RCRB and to permit the management of parameters that affect the behavior of Root Complex functionality associated with the RCRB.

Figure 7-202 *RCRB Header Extended Capability Structure*

### 7.9.7.1 RCRB Header Extended Capability Header (Offset 00h)

Figure 7-203 details allocation of register fields in the RCRB Header Extended Capability Header. Table 7-166 provides the respective bit definitions. Refer to Section 7.6.3 for a description of the PCI Express Enhanced Capabilities header. The Extended Capability ID for the RCRB Header Extended Capability is 000Ah.

Figure 7-203 *RCRB Header Extended Capability Header*Table 7-166 *RCRB Header Extended Capability Header*

Bit Location	Register Description	Attributes
15:0	<b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability. Extended Capability ID for the RCRB Header Extended Capability is 000Ah.	RO
19:16	<b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the Capability structure present. Must be 1h for this version of the specification.	RO
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Capability structure or 000h if no other items exist in the linked list of Capabilities. For Extended Capabilities implemented in Configuration Space, this offset is relative to the beginning of PCI-compatible Configuration Space and thus must always be either 000h (for terminating list of Capabilities) or greater than OFFh.	RO

### 7.9.7.2 RCRB Vendor ID and Device ID register (Offset 04h)

Figure 7-204 details allocation of register fields in the RCRB Vendor ID and Device ID register; Table 7-167 provides the respective bit definitions.



Figure 7-204 RCRB Vendor ID and Device ID register

Table 7-167 RCRB Vendor ID and Device ID register

Bit Location	Register Description	Attributes
15:0	<b>Vendor ID</b> - PCI-SIG assigned. Analogous to the equivalent field in PCI-compatible Configuration Space. This field provides a means to associate an RCRB with a particular vendor.	RO
31:16	<b>Device ID</b> - Vendor assigned. Analogous to the equivalent field in PCI-compatible Configuration Space. This field provides a means for a vendor to classify a particular RCRB.	RO

### 7.9.7.3 RCRB Capabilities register (Offset 08h)

Figure 7-205 details allocation of register fields in the RCRB Capabilities register; Table 7-168 provides the respective bit definitions.

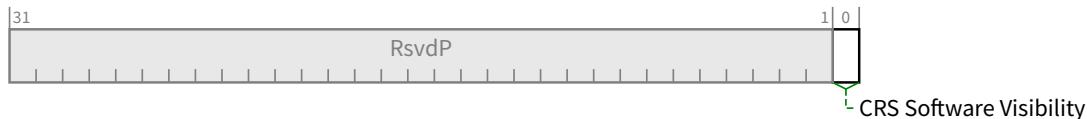


Figure 7-205 RCRB Capabilities register

Table 7-168 RCRB Capabilities register

Bit Location	Register Description	Attributes
0	<b>CRS Software Visibility</b> - When Set, this bit indicates that the Root Complex is capable of returning Configuration Request Retry Status (CRS) Completion Status to software for all Root Ports and integrated devices associated with this RCRB (see Section 2.3.1).	RO

### 7.9.7.4 RCRB Control register (Offset 0Ch)

Figure 7-206 details allocation of register fields in the RCRB Control register; Table 7-169 provides the respective bit definitions.

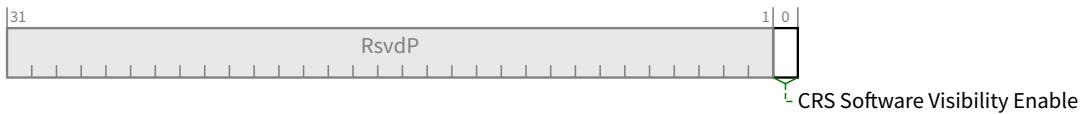


Figure 7-206 RCRB Control register

Table 7-169 RCRB Control register

Bit Location	Register Description	Attributes
0	<p><b>CRS Software Visibility Enable</b> - When Set, this bit enables the Root Complex to return Configuration Request Retry Status (CRS) Completion Status to software for all Root Ports and integrated devices associated with this RCRB (see Section 2.3.1 ).</p> <p>RCRBs that do not implement this capability must hardwire this bit to 0b.</p> <p>Default value of this bit is 0b.</p>	RW

## 7.9.8 Root Complex Link Declaration Extended Capability

The Root Complex Link Declaration Extended Capability is an optional Capability that is permitted to be implemented by Root Ports, RCiEPs, or RCRBs to declare a Root Complex's internal topology.

A Root Complex consists of one or more following elements:

- PCI Express Root Port
- A default system Egress Port or an internal sink unit such as memory (represented by an RCRB)
- Internal Data Paths/Links (represented by an RCRB on either side of an internal Link)
- Integrated devices
- Functions

A Root Complex Component is a logical aggregation of the above described Root Complex elements. No single element can be part of more than one Root Complex Component. Each Root Complex Component must have a unique Component ID.

A Root Complex is represented either as an opaque Root Complex or as a collection of one or more Root Complex Components.

The Root Complex Link Declaration Extended Capability is permitted to be present in a Root Complex element's Configuration Space or RCRB. It declares Links from the respective element to other elements of the same Root Complex Component or to an element in another Root Complex Component. The Links are required to be declared bidirectional such that each valid data path from one element to another has corresponding Link Entries in the Configuration Space (or RCRB) of both elements.

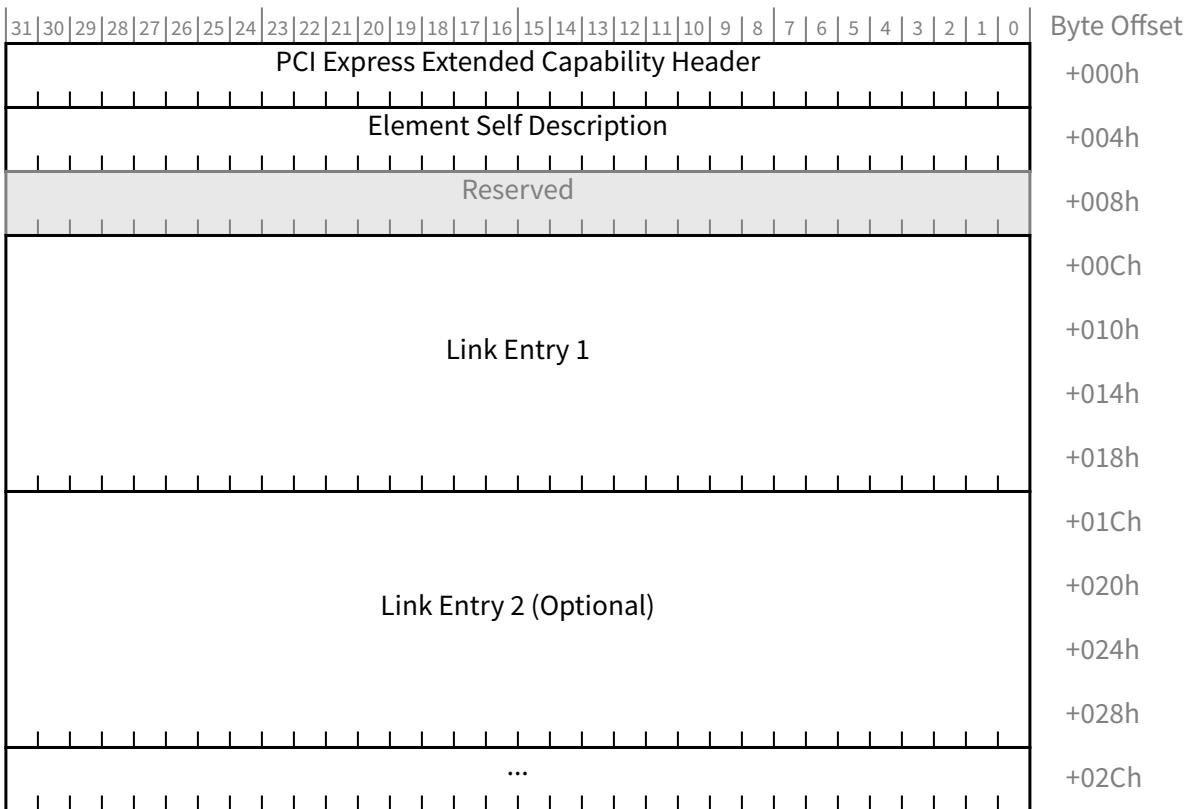
The Root Complex Link Declaration Extended Capability is permitted to also declare an association between a Configuration Space element (Root Port or RCiEP) and an RCRB Header Extended Capability (see Section 7.9.7) contained in an RCRB that affects the behavior of the Configuration Space element. Note that an RCRB Header association is not declared bidirectional; the association is only declared by the Configuration Space element and not by the target RCRB.

## IMPLEMENTATION NOTE

### Topologies to Avoid

Topologies that create more than one data path between any two Root Complex elements (either directly or through other Root Complex elements) may not be able to support bandwidth allocation in a standard manner. The description of how traffic is routed through such a topology is implementation specific, meaning that general purpose-operating systems may not have enough information about such a topology to correctly support bandwidth allocation. In order to circumvent this problem, these operating systems may require that a single RCRB element (of type Internal Link) not declare more than one Link to a Root Complex Component other than the one containing the RCRB element itself.

The Root Complex Link Declaration Extended Capability, as shown in [Figure 7-207](#), consists of the PCI Express Extended Capability header and Root Complex Element Self Description followed by one or more Root Complex Link Entries.



*Figure 7-207 Root Complex Link Declaration Extended Capability*

#### 7.9.8.1 Root Complex Link Declaration Extended Capability Header (Offset 00h)

The Extended Capability ID for the Root Complex Link Declaration Extended Capability is 0005h.

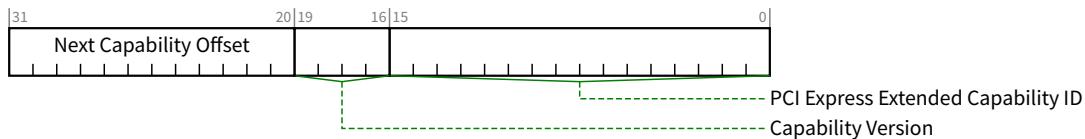


Figure 7-208 Root Complex Link Declaration Extended Capability Header

Table 7-170 Root Complex Link Declaration Extended Capability Header

Bit Location	Register Description	Attributes
15:0	<b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability.  The Extended Capability ID for the Root Complex Link Declaration Extended Capability is 0005h.	<u>RO</u>
19:16	<b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the Capability structure present.  Must be 1h for this version of the specification.	<u>RO</u>
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Capability structure or 000h if no other items exist in the linked list of Capabilities.  For Extended Capabilities implemented in Configuration Space, this offset is relative to the beginning of PCI-compatible Configuration Space and thus must always be either 000h (for terminating list of Capabilities) or greater than OFFh.  The bottom 2 bits of this offset are Reserved and must be implemented as 00b although software must mask them to allow for future uses of these bits.	<u>RO</u>

### 7.9.8.2 Element Self Description Register (Offset 04h)

The Element Self Description Register provides information about the Root Complex element containing the Root Complex Link Declaration Extended Capability.

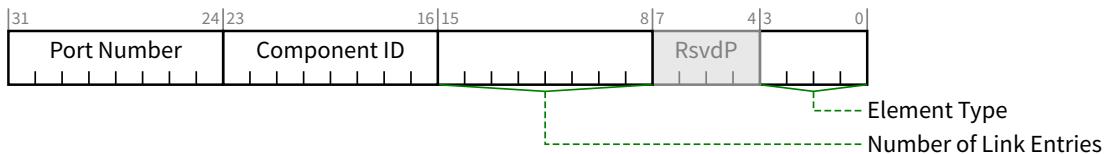


Figure 7-209 Element Self Description Register

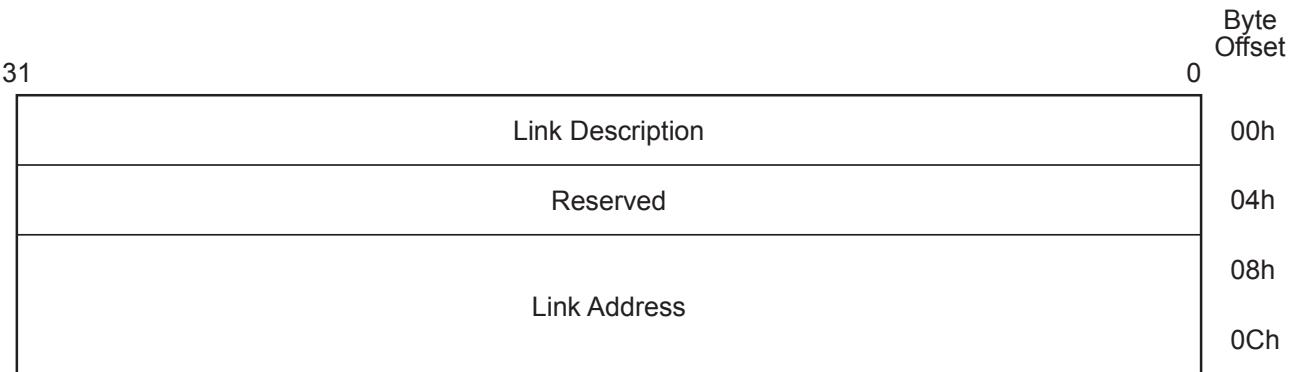
Table 7-171 Element Self Description Register

Bit Location	Register Description	Attributes
3:0	<b>Element Type</b> - This field indicates the type of the Root Complex Element. Defined encodings are:	<u>RO</u>

Bit Location	Register Description	Attributes
	<b>0h</b> Configuration Space Element <b>1h</b> System Egress Port or internal sink (memory) <b>2h</b> Internal Root Complex Link <b>3h-Fh</b> Reserved	
15:8	<b>Number of Link Entries</b> - This field indicates the number of <u>Link Entries</u> following the Element Self Description. This field must report a value of 01h or higher.	<u>HwInit</u>
23:16	<b>Component ID</b> - This field identifies the Root Complex Component that contains this Root Complex Element. Component IDs must start at 01h, as a value of 00h is Reserved.	<u>HwInit</u>
31:24	<b>Port Number</b> - This field specifies the Port Number associated with this element with respect to the Root Complex Component that contains this element.  An element with a Port Number of 00h indicates the default Egress Port to configuration software.	<u>HwInit</u>

### 7.9.8.3 Link Entries

Link Entries start at offset 10h of the Root Complex Link Declaration Extended Capability structure. Each Link Entry consists of a Link description followed by a 64-bit Link Address at offset 08h from the start of Link Entry identifying the target element for the declared Link. A Link Entry declares an internal Link to another Root Complex Element.



A-0420

Figure 7-210 Link Entry

#### 7.9.8.3.1 Link Description Register

The Link Description Register is located at offset 00h from the start of a Link Entry and is defined as follows:

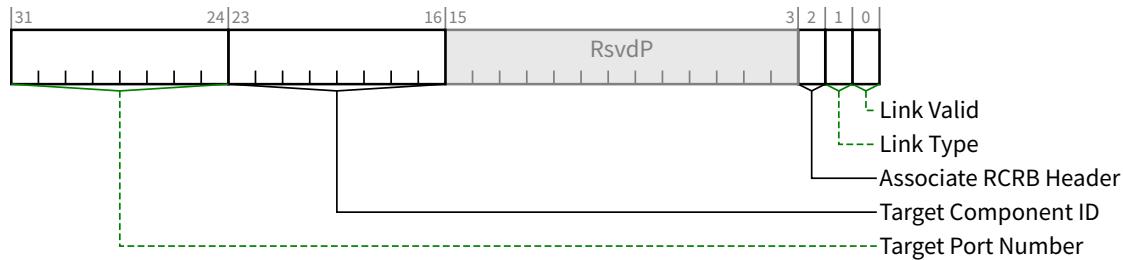


Figure 7-211 Link Description Register

Table 7-172 Link Description Register

Bit Location	Register Description	Attributes
0	<b>Link Valid</b> - When Set, this bit indicates that the <u>Link Entry</u> specifies a valid Link. <u>Link Entries</u> that do not have either this bit Set or the <u>Associate RCRB Header</u> bit Set (or both) are ignored by software.	<u>HwInit</u>
1	<b>Link Type</b> - This bit indicates the target type of the Link and defines the format of the <u>Link Address</u> field. Defined <u>Link Type</u> values are:  <b>0b</b> Link points to memory-mapped space <sup>156</sup> (for <u>RCRB</u> ). The <u>Link Address</u> specifies the 64-bit base address of the target <u>RCRB</u> .  <b>1b</b> Link points to Configuration Space (for a Root Port or <u>RCiEP</u> ). The <u>Link Address</u> specifies the configuration address (PCI Segment Group, Bus, Device, Function) of the target element.	<u>HwInit</u>
2	<b>Associate RCRB Header</b> - When Set, this bit indicates that the <u>Link Entry</u> associates the declaring element with an <u>RCRB Header Extended Capability</u> in the target <u>RCRB</u> . <u>Link Entries</u> that do not have either this bit Set or the <u>Link Valid</u> bit Set (or both) are ignored by software.  The <u>Link Type</u> bit must be Clear when this bit is Set.	<u>HwInit</u>
23:16	<b>Target Component ID</b> - This field identifies the Root Complex Component that is targeted by this <u>Link Entry</u> . Components IDs must start at 01h, as a value of 00h is Reserved	<u>HwInit</u>
31:24	<b>Target Port Number</b> - This field specifies the Port Number associated with the element targeted by this <u>Link Entry</u> ; the <u>Target Port Number</u> is with respect to the Root Complex Component (identified by the <u>Target Component ID</u> ) that contains the target element.	<u>HwInit</u>

### 7.9.8.3.2 Link Address

The Link Address is a HwInit field located at offset 08h from the start of a Link Entry that identifies the target element for the Link Entry. For a Link of Link Type 0 in its Link Description, the Link Address specifies the memory-mapped base address of RCRB. For a Link of Link Type 1 in its Link Description, the Link Address specifies the Configuration Space address of a PCI Express Root Port or an RCiEP.

<sup>156</sup>. The memory-mapped space for accessing an RCRB is not the same as Memory Space, and must not overlap with Memory Space.

### 7.9.8.3.2.1 Link Address for Link Type 0

For a Link pointing to a memory-mapped RCRB (Link Type bit = 0), the first DWORD specifies the lower 32 bits of the RCRB base address of the target element as shown below; bits 11:0 are hardwired to 000h and Reserved for future use. The second DWORD specifies the high order 32 bits (63:32) of the RCRB base address of the target element.

31		0	Byte Offset
	Link Description		00h
	Reserved		04h
	Link Address Bits 31:0		08h
	Link Address Bits 63:32		0Ch

A-0418

Figure 7-212 Link Address for Link Type 0

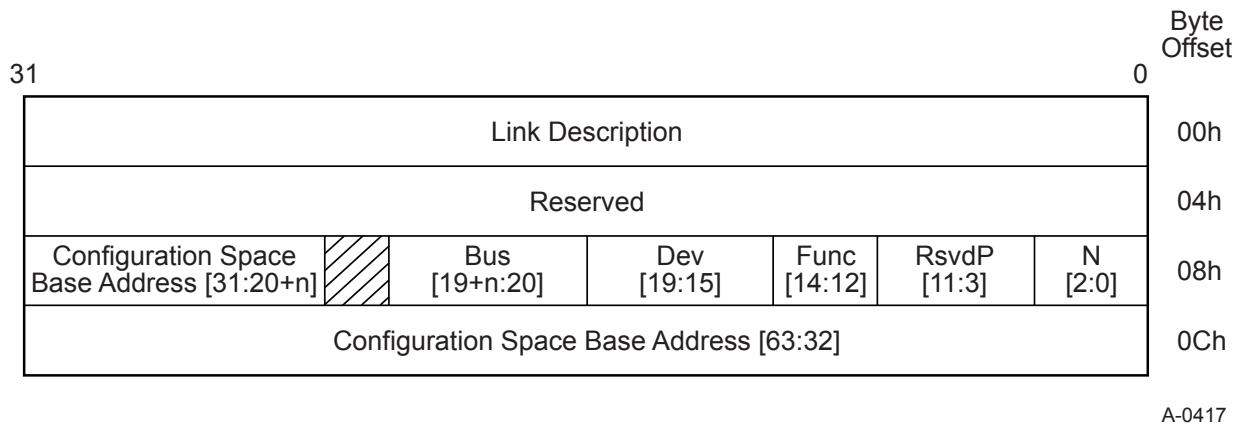
### 7.9.8.3.2.2 Link Address for Link Type 1

For a Link pointing to the Configuration Space of a Root Complex element (Link Type bit = 1), bits in the first DWORD specify the Bus, Device, and Function Number of the target element. As shown in Figure 7-213, bits 2:0 (N) encode the number of bits n associated with the Bus Number, with N = 000b specifying n = 8 and all other encodings specifying n = <value of N>. Bits 11:3 are Reserved and hardwired to 0. Bits 14:12 specify the Function Number, and bits 19:15 specify the Device Number. Bits (19 + n):20 specify the Bus Number, with  $1 \leq n \leq 8$ .

Bits 31:(20 + n) of the first DWORD together with the second DWORD optionally identify the target element's hierarchy for systems implementing the PCI Express Enhanced Configuration Access Mechanism by specifying bits 63:(20 + n) of the memory-mapped Configuration Space base address of the PCI Express hierarchy associated with the targeted element; single hierarchy systems that do not implement more than one memory mapped Configuration Space are allowed to report a value of zero to indicate default Configuration Space.

A Configuration Space base address [63:(20 + n)] equal to zero indicates that the Configuration Space address defined by bits (19 + n):12 (Bus Number, Device Number, and Function Number) exists in the default PCI Segment Group; any non-zero value indicates a separate Configuration Space base address.

Software must not use n outside the context of evaluating the Bus Number and memory-mapped Configuration Space base address for this specific target element. In particular, n does not necessarily indicate the maximum Bus Number supported by the associated PCI Segment Group.



A-0417

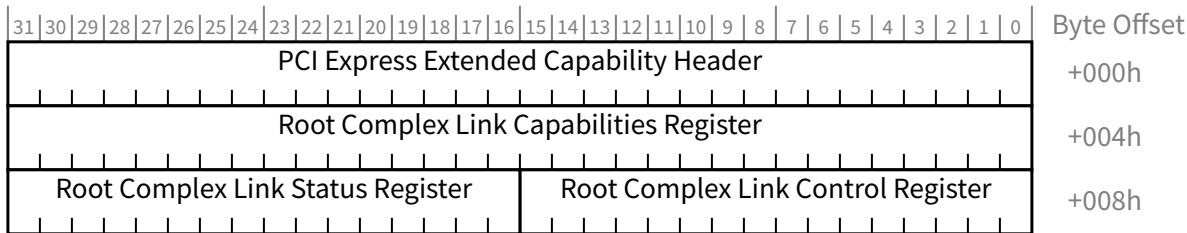
Figure 7-213 Link Address for Link Type 1Table 7-173 Link Address for Link Type 1

Bit Location	Register Description	Attributes
2:0	<b>N</b> - Encoded number of Bus Number bits	<u>HwInit</u>
14:12	<b>Function Number</b>	<u>HwInit</u>
19:15	<b>Device Number</b>	<u>HwInit</u>
(19 + n):20	<b>Bus Number</b>	<u>HwInit</u>
63:(20 + n)	<b>PCI Express Configuration Space Base Address</b> ( $1 \leq n \leq 8$ ) Note: A Root Complex that does not implement multiple Configuration Spaces is allowed to report this field as 0.	<u>HwInit</u>

### 7.9.9 Root Complex Internal Link Control Extended Capability

The Root Complex Internal Link Control Extended Capability is an optional Capability that controls an internal Root Complex Link between two distinct Root Complex Components. This Capability is valid for RCRBs that declare an Element Type field as Internal Root Complex Link in the Element Self-Description register of the Root Complex Link Declaration Capability structure.

The Root Complex Internal Link Control Extended Capability structure is defined as shown in Figure 7-214 .

Figure 7-214 Root Complex Internal Link Control Extended Capability

### 7.9.9.1 Root Complex Internal Link Control Extended Capability Header (Offset 00h)

The Extended Capability ID for the Root Complex Internal Link Control Extended Capability is 0006h.

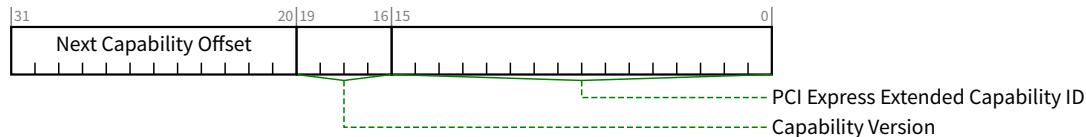
Figure 7-215 Root Complex Internal Link Control Extended Capability Header

Table 7-174 Root Complex Internal Link Control Extended Capability Header

Bit Location	Register Description	Attributes
15:0	<b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability. The Extended Capability ID for the <u>Root Complex Internal Link Control Extended Capability</u> is 0006h.	<u>RO</u>
19:16	<b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the Capability structure present. Must be 1h for this version of the specification.	<u>RO</u>
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Capability structure or 000h if no other items exist in the linked list of Capabilities. For Extended Capabilities implemented in Configuration Space, this offset is relative to the beginning of PCI-compatible Configuration Space and thus must always be either 000h (for terminating list of Capabilities) or greater than 0FFh. The bottom 2 bits of this offset are Reserved and must be implemented as 00b although software must mask them to allow for future uses of these bits.	<u>RO</u>

### 7.9.9.2 Root Complex Link Capabilities Register (Offset 04h)

The Root Complex Link Capabilities Register identifies capabilities for this Link.

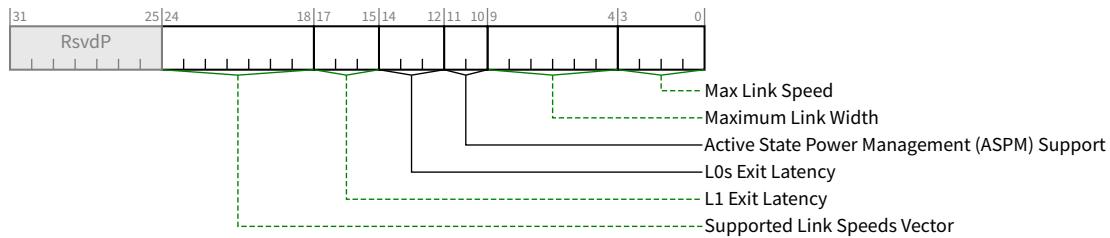


Figure 7-216 Root Complex Link Capabilities Register

Table 7-175 Root Complex Link Capabilities Register

Bit Location	Register Description	Attributes
3:0	<p><b>Max Link Speed</b> - This field indicates the maximum Link speed of the associated Link. The encoded value specifies a bit location in the <a href="#">Supported Link Speeds Vector</a> (in the <a href="#">Root Complex Link Capabilities Register</a>) that corresponds to the maximum Link speed.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>0001b</b> Supported Link Speeds Vector field bit 0</li> <li><b>0010b</b> Supported Link Speeds Vector field bit 1</li> <li><b>0011b</b> Supported Link Speeds Vector field bit 2</li> <li><b>0100b</b> Supported Link Speeds Vector field bit 3</li> <li><b>0101b</b> Supported Link Speeds Vector field bit 4</li> <li><b>0110b</b> Supported Link Speeds Vector field bit 5</li> <li><b>0111b</b> Supported Link Speeds Vector field bit 6</li> <li><b>Others</b> All other encodings are reserved.</li> </ul> <p>A Root Complex that does not support this feature must report 0000b in this field.</p>	<u>RO</u>
9:4	<p><b>Maximum Link Width</b> - This field indicates the maximum width of the given Link.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>00 0001b</b> x1</li> <li><b>00 0010b</b> x2</li> <li><b>00 0100b</b> x4</li> <li><b>00 1000b</b> x8</li> <li><b>00 1100b</b> x12</li> <li><b>01 0000b</b> x16</li> <li><b>10 0000b</b> x32</li> </ul> <p>All other encodings are Reserved. A Root Complex that does not support this feature must report 00 0000b in this field.</p>	<u>RO</u>
11:10	<p><b>Active State Power Management (ASPM) Support</b> - This field indicates the level of ASPM supported on the given Link.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>00b</b> No ASPM Support</li> <li><b>01b</b> L0s Supported</li> </ul>	<u>RO</u>

Bit Location	Register Description	Attributes
	<p><b>10b</b> L1 Supported</p> <p><b>11b</b> L0s and L1 Supported</p>	
14:12	<p><b>L0s Exit Latency</b> - This field indicates the L0s exit latency for the given Link. The value reported indicates the length of time this Port requires to complete transition from L0s to L0. If L0s is not supported, the value is undefined.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>000b</b> Less than 64 ns</li> <li><b>001b</b> 64 ns to less than 128 ns</li> <li><b>010b</b> 128 ns to less than 256 ns</li> <li><b>011b</b> 256 ns to less than 512 ns</li> <li><b>100b</b> 512 ns to less than 1 <math>\mu</math>s</li> <li><b>101b</b> 1 <math>\mu</math>s to less than 2 <math>\mu</math>s</li> <li><b>110b</b> 2 <math>\mu</math>s to 4 <math>\mu</math>s</li> <li><b>111b</b> More than 4 <math>\mu</math>s</li> </ul>	RO
17:15	<p><b>L1 Exit Latency</b> - This field indicates the L1 exit latency for the given Link. The value reported indicates the length of time this Port requires to complete transition from ASPM L1 to L0. If ASPM L1 is not supported, the value is undefined.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>000b</b> Less than 1 <math>\mu</math>s</li> <li><b>001b</b> 1 <math>\mu</math>s to less than 2 <math>\mu</math>s</li> <li><b>010b</b> 2 <math>\mu</math>s to less than 4 <math>\mu</math>s</li> <li><b>011b</b> 4 <math>\mu</math>s to less than 8 <math>\mu</math>s</li> <li><b>100b</b> 8 <math>\mu</math>s to less than 16 <math>\mu</math>s</li> <li><b>101b</b> 16 <math>\mu</math>s to less than 32 <math>\mu</math>s</li> <li><b>110b</b> 32 <math>\mu</math>s to 64 <math>\mu</math>s</li> <li><b>111b</b> More than 64 <math>\mu</math>s</li> </ul>	RO
24:18	<p><b>Supported Link Speeds Vector</b> - This field indicates the supported Link speed(s) of the associated Link. For each bit, a value of 1b indicates that the corresponding Link speed is supported; otherwise, the Link speed is not supported. See <a href="#">Section 8.2.1</a> for further requirements.</p> <p>Bit definitions within this field are:</p> <ul style="list-style-type: none"> <li><b>Bit 0</b> 2.5 GT/s</li> <li><b>Bit 1</b> 5.0 GT/s</li> <li><b>Bit 2</b> 8.0 GT/s</li> <li><b>Bit 3</b> 16.0 GT/s</li> <li><b>Bit 4</b> 32.0 GT/s</li> <li><b>Bits 6:5</b> RsvdP</li> </ul>	RO

## IMPLEMENTATION NOTE

### Supported Link Speeds With Earlier Hardware

Hardware components compliant to versions prior to the [PCIe-3.0] did not implement the Supported Link Speeds Vector field and instead returned 0000 000b in bits 24:18.

For software to determine the supported Link speeds for components where this field is contains 0000 000b, software can read bits 3:0 of the Root Complex Link Capabilities Register (now defined to be the Max Link Speed field), and interpret the value as follows:

**0001b**

2.5 GT/s Link speed supported

**0010b**

5.0 GT/s and 2.5 GT/s Link speeds supported

For such components, the same encoding is also used for the values for the Current Link Speed field (in the Root Complex Link Status Register).

## IMPLEMENTATION NOTE

### Software Management of Link Speeds With Future Hardware

It is strongly encouraged that software primarily utilize the Supported Link Speeds Vector instead of the Max Link Speed field, so that software can determine the exact set of supported speeds on current and future hardware. This can avoid software being confused if a future specification defines Links that do not require support for all slower speeds.

#### 7.9.9.3 Root Complex Link Control Register (Offset 08h)

The Root Complex Link Control Register controls parameters for this internal Link.

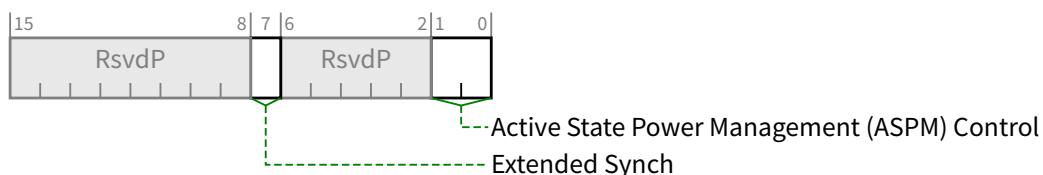


Figure 7-217 Root Complex Link Control Register

Table 7-176 Root Complex Link Control Register

Bit Location	Register Description	Attributes
1:0	<p><b>Active State Power Management (ASPM) Control</b> - This field controls the level of ASPM enabled on the given Link.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>00b</b> Disabled</li> <li><b>01b</b> L0s Entry Enabled</li> <li><b>10b</b> L1 Entry Enabled</li> <li><b>11b</b> L0s and L1 Entry Enabled</li> </ul> <p>Note: “L0s Entry Enabled” enables the Transmitter to enter L0s. If L0s is supported, the Receiver must be capable of entering L0s even when the Transmitter is disabled from entering L0s (00b or 10b).</p> <p>Default value of this field is implementation specific.</p> <p>Software must not enable L0s in either direction on a given Link unless components on both sides of the Link each support L0s, as indicated by their ASPM Support field values. Otherwise, the result is undefined.</p> <p>ASPM L1 must be enabled by software in the Upstream component on a Link prior to enabling ASPM L1 in the Downstream component on that Link. When disabling ASPM L1, software must disable ASPM L1 in the Downstream component on a Link prior to disabling ASPM L1 in the Upstream component on that Link. ASPM L1 must only be enabled on the Downstream component if both components on a Link support ASPM L1.</p> <p>A Root Complex that does not support this feature for the given internal Link must hardwire this field to 00b.</p>	RW
7	<p><b>Extended Synch</b> - This bit when Set forces the transmission of additional Ordered Sets when exiting the L0s state (see Section 4.2.4.6) and when in the Recovery state (see Section 4.2.6.4.1). This mode provides external devices (e.g., logic analyzers) monitoring the Link time to achieve bit and Symbol lock before the Link enters the L0 state and resumes communication.</p> <p>A Root Complex that does not support this feature for the given internal Link must hardwire this bit to 0b.</p> <p>Default value for this bit is 0b.</p>	RW

#### 7.9.9.4 Root Complex Link Status Register (Offset 0Ah)

The Root Complex Link Status Register provides information about Link specific parameters.

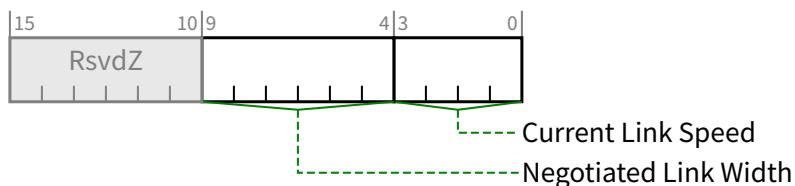


Figure 7-218 Root Complex Link Status Register

*Table 7-177 Root Complex Link Status Register*

Bit Location	Register Description	Attributes
3:0	<p><b>Current Link Speed</b> - This field indicates the negotiated Link speed of the given Link. The encoded value specifies a bit location in the <u>Supported Link Speeds Vector</u> (in the <u>Root Complex Link Capabilities Register</u>) that corresponds to the current Link speed.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>0001b</b> Supported Link Speeds Vector field bit 0</li> <li><b>0010b</b> Supported Link Speeds Vector field bit 1</li> <li><b>0011b</b> Supported Link Speeds Vector field bit 2</li> <li><b>0100b</b> Supported Link Speeds Vector field bit 3</li> <li><b>0101b</b> Supported Link Speeds Vector field bit 4</li> <li><b>0110b</b> Supported Link Speeds Vector field bit 5</li> <li><b>0111b</b> Supported Link Speeds Vector field bit 6</li> </ul> <p>All other encodings are Reserved.</p> <p>The value in this field is undefined when the Link is not up. A Root Complex that does not support this feature must report 0000b in this field.</p>	RO
9:4	<p><b>Negotiated Link Width</b> - This field indicates the negotiated width of the given Link.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>00 0001b</b> x1</li> <li><b>00 0010b</b> x2</li> <li><b>00 0100b</b> x4</li> <li><b>00 1000b</b> x8</li> <li><b>00 1100b</b> x12</li> <li><b>01 0000b</b> x16</li> <li><b>10 0000b</b> x32</li> </ul> <p>All other encodings are Reserved. The value in this field is undefined when the Link is not up. A Root Complex that does not support this feature must hardwire this field to 00 0000b.</p>	RO

## 7.9.10 Root Complex Event Collector Endpoint Association Extended Capability

The Root Complex Event Collector Endpoint Association Extended Capability is implemented by Root Complex Event Collectors. It declares the RCiEPs supported by the Root Complex Event Collector. A Root Complex Event Collector must implement the Root Complex Event Collector Endpoint Association Extended Capability; no other PCI Express Device Function is permitted to implement this Capability.

The Root Complex Event Collector Endpoint Association Extended Capability, as shown in [Figure 7-219](#), consists of the PCI Express Extended Capability header followed by a DWORD bitmap enumerating RCiEPs on the same Bus, and optionally an additional range of Bus Numbers that may contain RCiEPs associated with the Root Complex Event Collector. Functions other than RCiEPs (e.g. Root Ports) contained in the range described by this Capability are not associated with this Root Complex Event Collector.

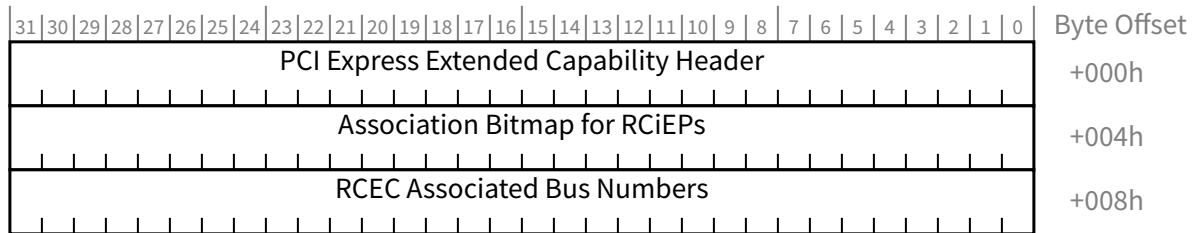


Figure 7-219 Root Complex Event Collector Endpoint Association Extended Capability

### 7.9.10.1 Root Complex Event Collector Endpoint Association Extended Capability Header (Offset 00h)

The Extended Capability ID for the Root Complex Event Collector Endpoint Association Extended Capability is 0007h. Figure 7-220 details allocation of fields in the Root Complex Event Collector Endpoint Association Extended Capability Header; Table 7-178 provides the respective bit definitions.

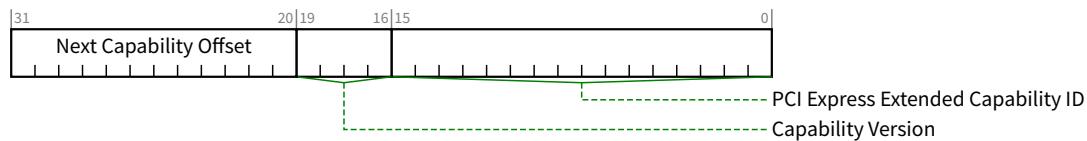


Figure 7-220 Root Complex Event Collector Endpoint Association Extended Capability Header

Table 7-178 Root Complex Event Collector Endpoint Association Extended Capability Header

Bit Location	Register Description	Attributes
15:0	<b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability.  The Extended Capability ID for the Root Complex Event Collector Endpoint Association Extended Capability is 0007h.	RO
19:16	<b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the Capability structure present.  Must be 2h if the Extended Capability contains the RCEC Associated Bus Numbers Register (see Section 7.9.10.3). Must be 1h otherwise.	RO
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Capability structure or 000h if no other items exist in the linked list of Capabilities.  For Extended Capabilities implemented in Configuration Space, this offset is relative to the beginning of PCI-compatible Configuration Space and thus must always be either 000h (for terminating list of Capabilities) or greater than OFFh.  The bottom 2 bits of this offset are Reserved and must be implemented as 00b although software must mask them to allow for future uses of these bits.	RO

### 7.9.10.2 Association Bitmap for RCiEPs (Offset 04h)

The Association Bitmap for RCiEPs is a read-only register that sets the bits corresponding to the Device Numbers of RCiEPs associated with the Root Complex Event Collector on the same Bus Number as the Event Collector itself. The bit corresponding to the Device Number of the Root Complex Event Collector must always be Set.

### 7.9.10.3 RCEC Associated Bus Numbers Register (Offset 08h)

The RCEC Associated Bus Numbers Register is a read-only register that indicates an additional range of Bus Numbers containing RCiEPs associated with this Root Complex Event Collector. It is permitted for Functions other than RCiEPs, including Root Ports, to appear within the Association Bus Range. Only RCiEPs in the range are associated with this Root Complex Event Collector. This register is present if the Capability Version is 2h or greater.

This register does not indicate association between an Event Collector and any Virtual Functions within the Association Bus Range (see Section 9.2.1.2). This register does not indicate association between an Event Collector and any Function on the same Bus Number as the Event Collector itself, however it is permitted for the Association Bus Range to include the Bus Number of the Root Complex Event Collector.

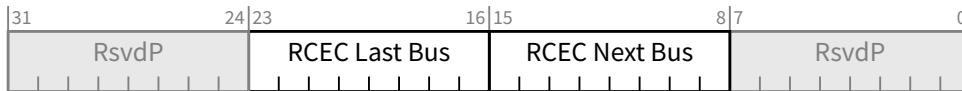


Figure 7-221 RCEC Associated Bus Numbers Register

Table 7-179 RCEC Associated Bus Numbers Register

Bit Location	Register Description	Attributes
15:8	<b>RCEC Next Bus</b> - This field contains the lowest additional bus number containing RCiEPs associated with this Root Complex Event Collector. If all of the Devices associated with this Root Complex Event Collector are on the same bus as the Event Collector, then this field must be set to FFh.	<u>HwInit</u>
23:16	<b>RCEC Last Bus</b> - This field contains the highest additional bus number containing RCiEPs associated with this Root Complex Event Collector.  If all of the Devices associated with this Root Complex Event Collector are on the same bus as the Event Collector, then this field must be set to 00h.	<u>HwInit</u>

## IMPLEMENTATION NOTE

### RCEC Associated Bus Number Compatibility with Legacy Software

Legacy software may not support the use of the RCEC Associated Bus Numbers Register as a mechanism to associate Devices with a RCEC. Such software may see events in the RCEC from Devices on different bus numbers that it does not consider to be associated with the Root Complex Event Collector. System Software is strongly encouraged to report all events seen on the Root Complex Event Collector, regardless of whether or not it can determine association.

## 7.9.11 Multicast Extended Capability

Multicast is an optional normative functionality that is controlled by the Multicast Extended Capability structure. The Multicast Extended Capability is applicable to Root Ports, RCRBs, Switch Ports, Endpoint Functions, and RCiEPs. It is not applicable to PCI Express to PCI/PCI-X Bridges.

In the cases of a Switch or Root Complex or a component that contains multiple Functions, multiple copies of this Capability structure are required - one for each Endpoint Function, Switch Port, or Root Port that supports Multicast. To provide implementation efficiencies, certain fields within each of the Multicast Extended Capability structures within a component must be programmed the same and results are indeterminate if this is not the case. The fields and registers that must be configured with the same values include MC\_Enable, MC\_Num\_Group, MC\_Base\_Address and MC\_Index\_Position. These same fields in an Endpoint's Multicast Extended Capability structure must match those configured into a Multicast Extended Capability structure of the Switch or Root Complex above the Endpoint or in which the RCiEP is integrated.

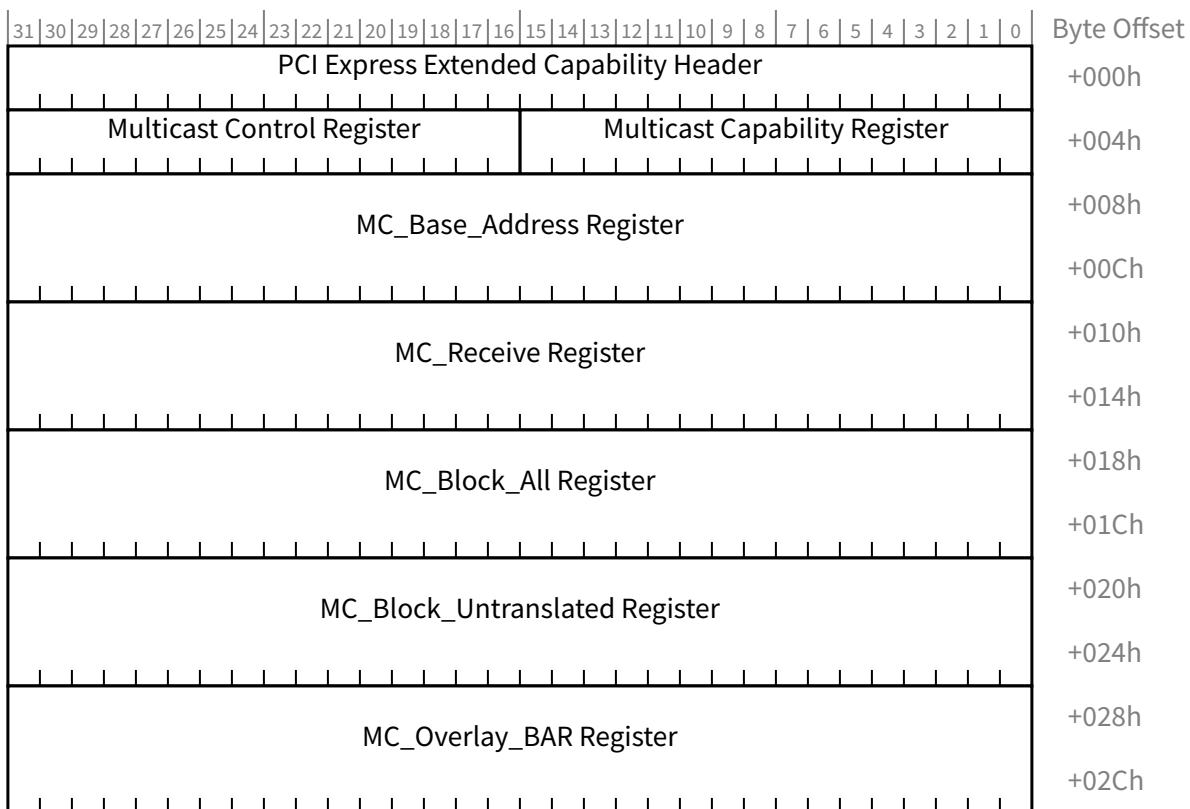


Figure 7-222 Multicast Extended Capability Structure

### 7.9.11.1 Multicast Extended Capability Header (Offset 00h)

Figure 7-223 details allocation of the fields in the Multicast Extended Capability Header and Table 7-180 provides the respective bit definitions.

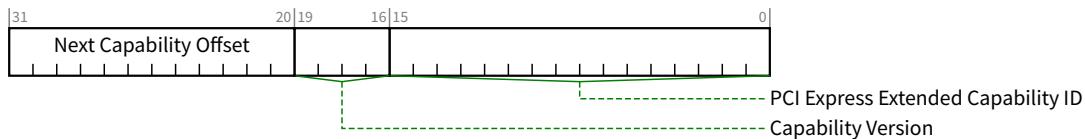


Figure 7-223 Multicast Extended Capability Header

Table 7-180 Multicast Extended Capability Header

Bit Location	Register Description	Attributes
15:0	<b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability. PCI Express Extended Capability ID for the <u>Multicast Extended Capability</u> is 0012h.	<u>RO</u>
19:16	<b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the Capability structure present. Must be 1h for this version of the specification.	<u>RO</u>
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Extended Capability structure or 000h if no other items exist in the linked list of Capabilities.	<u>RO</u>

### 7.9.11.2 Multicast Capability Register (Offset 04h)

Figure 7-224 details allocation of the fields in the Multicast Capability Register and Table 7-181 provides the respective bit definitions.

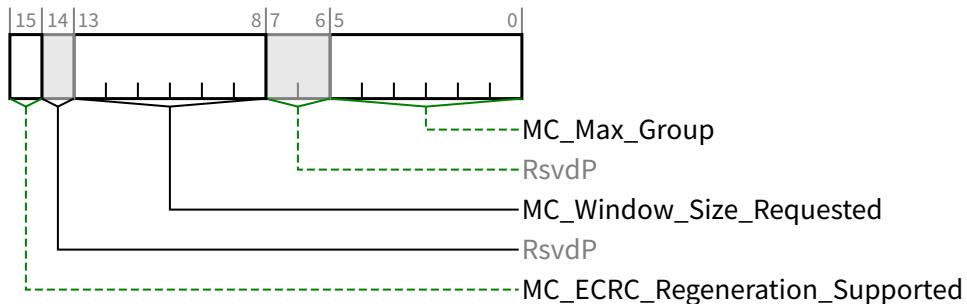


Figure 7-224 Multicast Capability Register

Table 7-181 Multicast Capability Register

Bit Location	Register Description	Attributes
5:0	<b>MC_Max_Group</b> - Value indicates the maximum number of Multicast Groups that the component supports, encoded as M-1. A value of 00h indicates that one Multicast Group is supported.	<u>RO</u>

Bit Location	Register Description	Attributes
13:8	<b>MC_Window_Size_Requested</b> - In Endpoints, the log <sub>2</sub> of the Multicast Window size requested. <u>RsvdP</u> in Switch and Root Ports.	RO
15	<b>MC_ECRC_Regeneration_Supported</b> - If Set, indicates that ECRC regeneration is supported.  This bit must not be Set unless the Function supports Advanced Error Reporting, and the ECRC Check Capable bit in the Advanced Error Capabilities and Control register is also Set. However, if ECRC regeneration is supported, its operation is not contingent upon the setting of the ECRC Check Enable bit in the Advanced Error Capabilities and Control register. This bit is applicable to Switch and Root Ports and is <u>RsvdP</u> in all other Functions.	RO/ <u>RsvdP</u>

### 7.9.11.3 Multicast Control Register (Offset 06h)

Table 7-182 details allocation of the fields in the Multicast Control Register and Table 7-182 provides the respective bit definitions.

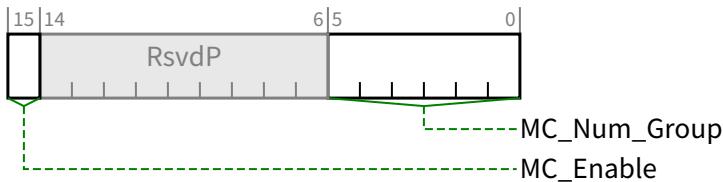


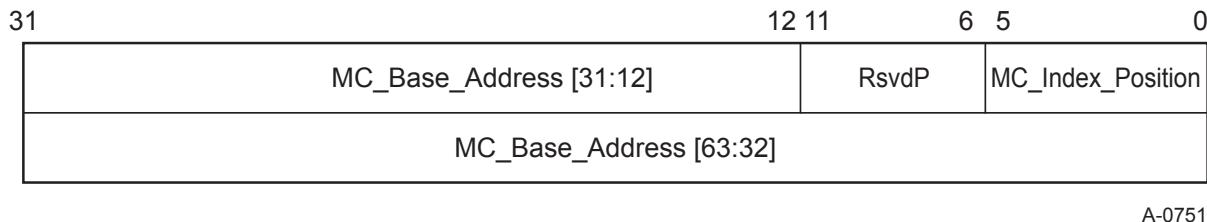
Figure 7-225 Multicast Control Register

Table 7-182 Multicast Control Register

Bit Location	Register Description	Attributes
5:0	<b>MC_Num_Group</b> - Value indicates the number of Multicast Groups configured for use, encoded as N-1. The default value of 00 0000b indicates that one Multicast Group is configured for use. Behavior is undefined if value exceeds MC_Max_Group. This parameter indirectly defines the upper limit of the Multicast address range. This field is ignored if MC_Enable is Clear. Default value is 00 0000b.	<u>RW</u>
15	<b>MC_Enable</b> - When Set, the Multicast mechanism is enabled for the component. Default value is 0b.	RW

### 7.9.11.4 MC\_Base\_Address Register (Offset 08h)

The MC\_Base\_Address Register contains the MC\_Base\_Address and the MC\_Index\_Position. Figure 7-226 details allocation of the fields in the MC\_Base\_Address Register and Table 7-183 provides the respective bit definitions.



A-0751

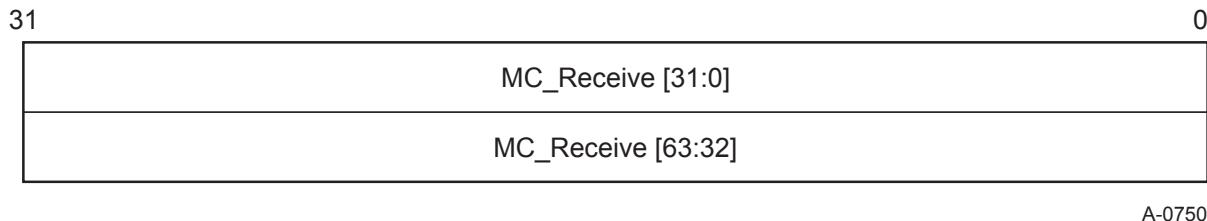
*Figure 7-226 MC\_Base\_Address Register**Table 7-183 MC\_Base\_Address Register*

Bit Location	Register Description	Attributes
5:0	<b>MC_Index_Position</b> - The location of the LSB of the Multicast Group number within the address. Behavior is undefined if this value is less than 12 and MC_Enable is Set. Default is 0.	RW
63:12	<b>MC_Base_Address</b> - The base address of the Multicast address range. The behavior is undefined if MC_Enable is Set and bits in this field corresponding to address bits that contain the Multicast Group number or address bits less than <u>MC_Index_Position</u> are non-zero. Default is 0.	RW

### 7.9.11.5 MC\_Receive Register (Offset 10h)

The MC\_Receive Register provides a bit vector denoting which Multicast groups the Function should accept, or in the case of Switch and Root Complex Ports, forward Multicast TLPs. This register is required in all Functions that implement the MC Capability structure.

Figure 7-227 details allocation of the fields in the MC\_Receive Register and Table 7-184 provides the respective bit definitions.



A-0750

*Figure 7-227 MC\_Receive Register**Table 7-184 MC\_Receive Register*

Bit Location	Register Description	Attributes
MC_Max_Group:0	<b>MC_Receive</b> - For each bit that's Set, this Function gets a copy of any Multicast TLPs for the associated Multicast Group. Bits above <u>MC_Num_Group</u> are ignored by hardware. Default value of each bit is 0b.	RW
All other bits	Reserved	RsvdP

### 7.9.11.6 MC\_Block\_All Register (Offset 18h)

The MC\_Block\_All Register provides a bit vector denoting which Multicast groups the Function should block. This register is required in all Functions that implement the MC Capability structure.

Figure 7-228 details allocation of the fields in the MC\_Block\_All Register and Table 7-185 provides the respective bit definitions.

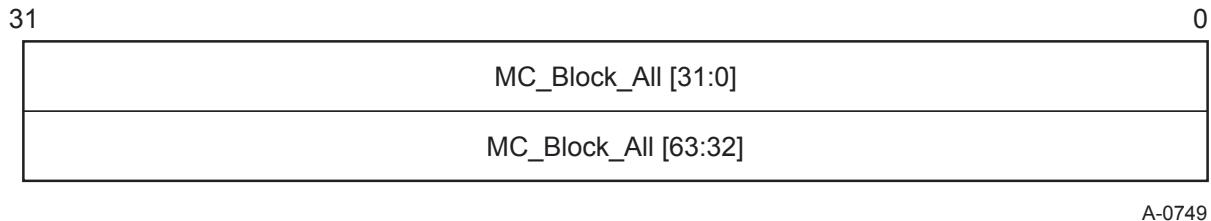


Figure 7-228 MC\_Block\_All Register

Table 7-185 MC\_Block\_All Register

Bit Location	Register Description	Attributes
<u>MC_Max_Group:0</u>	<b>MC_Block_All</b> - For each bit that is Set, this Function is blocked from sending TLPs to the associated Multicast Group. Bits above <u>MC_Num_Group</u> are ignored by hardware. Default value of each bit is 0b.	<u>RW</u>
All other bits	Reserved	<u>RsvdP</u>

### 7.9.11.7 MC\_Block\_Untranslated Register (Offset 20h)

The MC\_Block\_Untranslated Register is used to determine whether or not a TLP that includes an Untranslated Address should be blocked. This register is required in all Functions that implement the MC Capability structure. However, an Endpoint Function that does not implement the ATS capability may implement this register as RsvdP.

Figure 7-229 details allocation of the fields in the MC\_Block\_Untranslated Register and Table 7-186 provides the respective bit definitions.

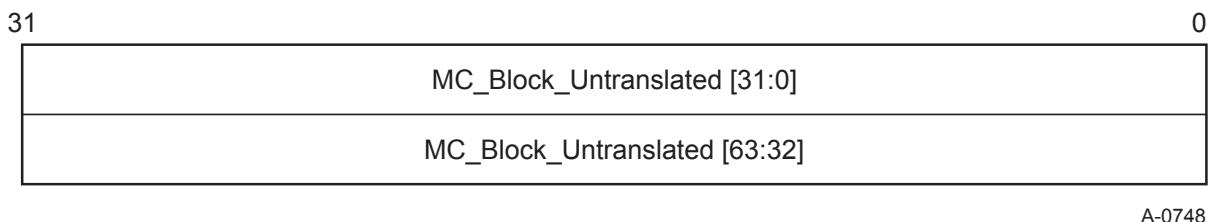


Figure 7-229 MC\_Block\_Untranslated Register

*Table 7-186 MC\_Block\_Untranslated Register*

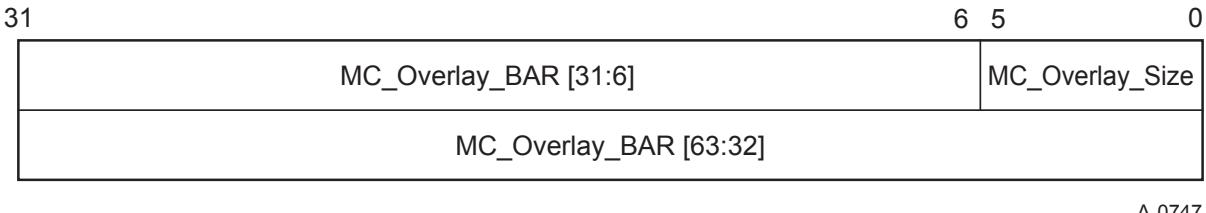
Bit Location	Register Description	Attributes
MC_Max_Group:0	<b>MC_Block_Untranslated</b> - For each bit that is Set, this Function is blocked from sending TLPs containing Untranslated Addresses to the associated MCG. Bits above <u>MC_Num_Group</u> are ignored by hardware. Default value of each bit is 0b.	RW
All other bits	Reserved	RsvdP

### 7.9.11.8 MC\_Overlay\_BAR Register (Offset 28h)

The MC\_Overlay\_BAR Register is required in Switch and Root Complex Ports that support the Multicast Extended Capability and not implemented in Endpoints. Software must interpret the Device/Port Type field in the PCI Express Capabilities Register to determine if the MC\_Overlay\_BAR Register is present in a Function.

The MC\_Overlay\_BAR specifies the base address of a window in unicast space onto which Multicast TLPs going out an Egress Port are overlaid by a process of address replacement. This allows a single BAR in an Endpoint attached to the Switch or Root Port to be used for both unicast and Multicast traffic. At a Switch Upstream Port, it allows the Multicast address range, or a portion of it, to be overlayed onto host memory.

Figure 7-230 details allocation of the fields in the MC\_Overlay\_BAR Register and Table 7-187 provides the respective bit definitions.



A-0747

*Figure 7-230 MC\_Overlay\_BAR Register**Table 7-187 MC\_Overlay\_BAR Register*

Bit Location	Register Description	Attributes
5:0	<b>MC_Overlay_Size</b> - If 6 or greater, specifies the size in bytes of the overlay aperture as a power of 2. If less than 6, disables the overlay mechanism. Default value is 00 0000b.	RW
63:6	<b>MC_Overlay_BAR</b> - Specifies the base address of the window onto which MC TLPs passing through this Function will be overlaid. Default value is 0.	RW

### 7.9.12 Dynamic Power Allocation Extended Capability (DPA Capability)

The DPA Capability structure is shown in Figure 7-231.

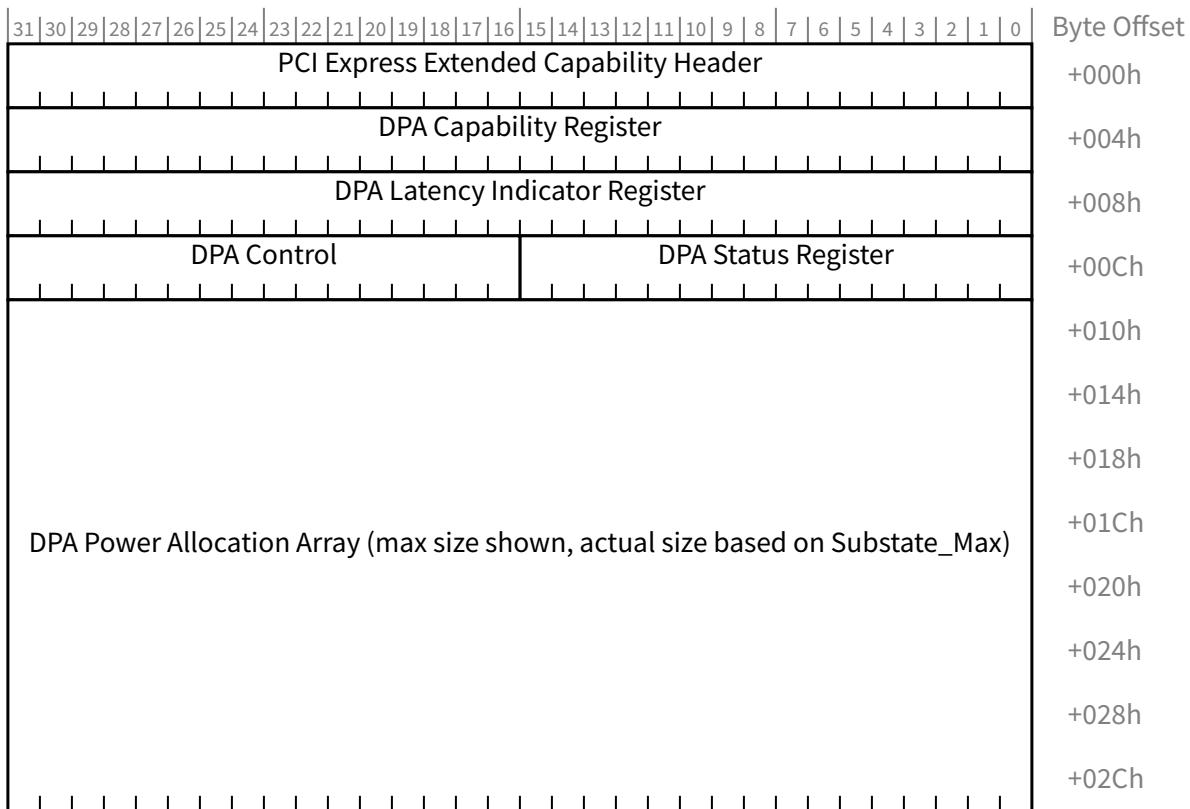


Figure 7-231 *Dynamic Power Allocation Extended Capability Structure*

### 7.9.12.1 DPA Extended Capability Header (Offset 00h)

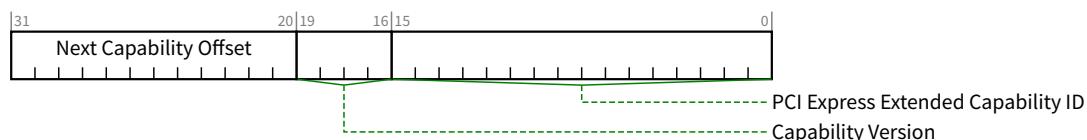


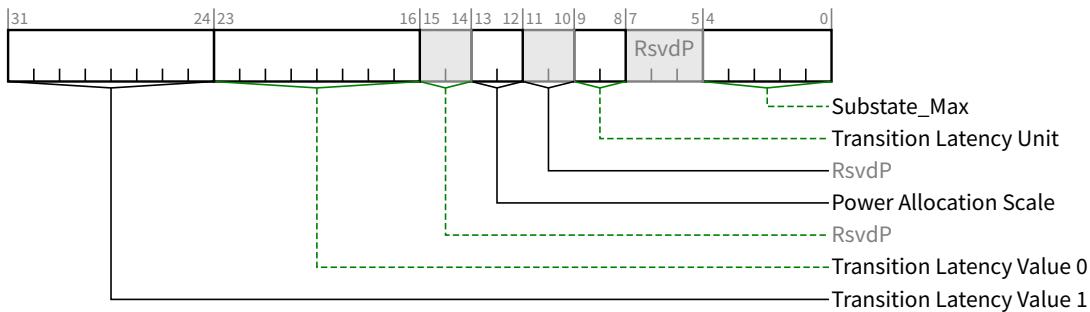
Figure 7-232 *DPA Extended Capability Header*

Table 7-188 *DPA Extended Capability Header*

Bit Location	Register Description	Attributes
15:0	<b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability. PCI Express Extended Capability ID for the DPA Extended Capability is 0016h.	RO
19:16	<b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the Capability structure present.	RO

Bit Location	Register Description	Attributes
	Must be 1h for this version of the specification.	
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Extended Capability structure or 000h if no other items exist in the linked list of Capabilities.	RO

#### **7.9.12.2 DPA Capability Register (Offset 04h)**



*Figure 7-233 DPA Capability Register*

*Table 7-189 DPA Capability Register*

Bit Location	Register Description	Attributes								
4:0	<b>Substate_Max</b> - Value indicates the maximum substate number, which is the total number of supported substates minus one. A value of 0 0000b indicates support for one substate.	<u>RO</u>								
9:8	<p><b>Transition Latency Unit (Tlunit)</b> - A substate's Transition Latency Value is multiplied by the <u>Transition Latency Unit</u> to determine the maximum Transition Latency for the substate.</p> <p>Defined encodings are</p> <table> <tr> <td><b>00b</b></td><td>1 ms</td></tr> <tr> <td><b>01b</b></td><td>10 ms</td></tr> <tr> <td><b>10b</b></td><td>100 ms</td></tr> <tr> <td><b>11b</b></td><td>Reserved</td></tr> </table>	<b>00b</b>	1 ms	<b>01b</b>	10 ms	<b>10b</b>	100 ms	<b>11b</b>	Reserved	<u>RO</u>
<b>00b</b>	1 ms									
<b>01b</b>	10 ms									
<b>10b</b>	100 ms									
<b>11b</b>	Reserved									
13:12	<p><b>Power Allocation Scale (PAS)</b> - The encodings provide the scale to determine power allocation per substate in Watts. The value corresponding to the substate in the <u>Substate Power Allocation</u> field is multiplied by this field to determine the power allocation for the substate.</p> <p>Defined encodings are</p> <table> <tr> <td><b>00b</b></td><td>10.0x</td></tr> <tr> <td><b>01b</b></td><td>1.0x</td></tr> <tr> <td><b>10b</b></td><td>0.1x</td></tr> <tr> <td><b>11b</b></td><td>0.01x</td></tr> </table>	<b>00b</b>	10.0x	<b>01b</b>	1.0x	<b>10b</b>	0.1x	<b>11b</b>	0.01x	<u>RO</u>
<b>00b</b>	10.0x									
<b>01b</b>	1.0x									
<b>10b</b>	0.1x									
<b>11b</b>	0.01x									
23:16	<b>Transition Latency Value 0 (Xlcy0)</b> - This value is multiplied by the <u>Transition Latency Unit</u> to determine the maximum Transition Latency for the substate	<u>RO</u>								

Bit Location	Register Description	Attributes
31:24	<b>Transition Latency Value 1 (XlcY1)</b> - This value is multiplied by the <u>Transition Latency Unit</u> to determine the maximum Transition Latency for the substate.	RO

### 7.9.12.3 DPA Latency Indicator Register (Offset 08h)

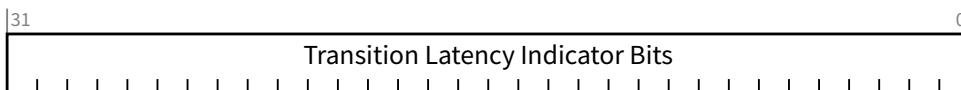


Figure 7-234 DPA Latency Indicator Register

Table 7-190 DPA Latency Indicator Register

Bit Location	Register Description	Attributes
31:0	<b>Transition Latency Indicator Bits</b> - Each bit indicates which Transition Latency Value is associated with the corresponding substate. A value of 0b indicates <u>Transition Latency Value 0</u> ; a value of 1b indicates <u>Transition Latency Value 1</u> . Only bits [Substate_Max:0] are defined. Bits above Substate_Max are RsvdP.	RO

### 7.9.12.4 DPA Status Register (Offset 0Ch)

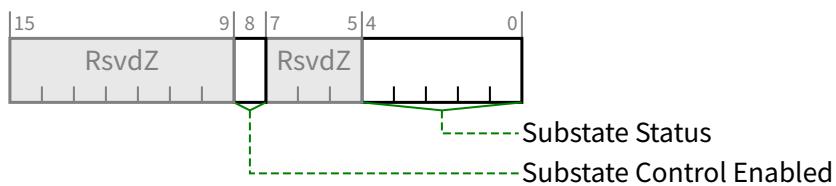


Figure 7-235 DPA Status Register

Table 7-191 DPA Status Register

Bit Location	Register Description	Attributes
4:0	<b>Substate Status</b> - Indicates current substate for this Function. Default is 0 0000b.	RO
8	<b>Substate Control Enabled</b> - Used by software to disable the <u>Substate Control</u> field in the DPA Control Register. Hardware sets this bit following a Conventional Reset or FLR. Software clears this bit by writing a 1b to it. Software is unable to set this bit directly. When this bit is Set, the <u>Substate Control</u> field determines the current substate. When this bit is Clear, the <u>Substate Control</u> field has no effect on the current substate.	RW1C

Bit Location	Register Description	Attributes
	Default value is 1b.	

### 7.9.12.5 DPA Control Register (Offset 0Eh)

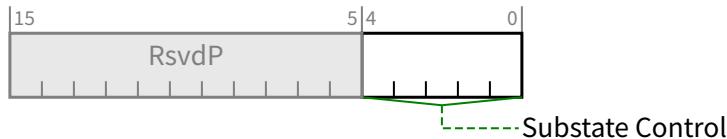
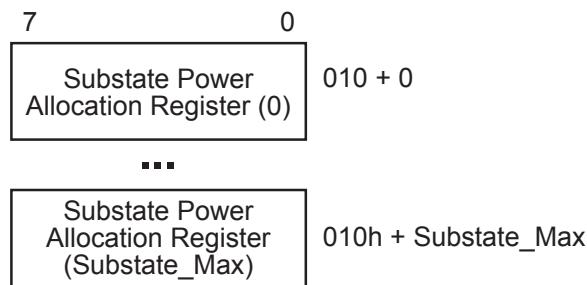


Figure 7-236 DPA Control Register

Table 7-192 DPA Control Register

Bit Location	Register Description	Attributes
4:0	<p><b>Substate Control</b> - Used by software to configure the Function substate. Software writes the substate value in this field to initiate a substate transition.</p> <p>When the <u>Substate Control Enabled</u> bit in the <u>DPA Status Register</u> is Set, this field determines the Function substate.</p> <p>When the <u>Substate Control Enabled</u> bit in the <u>DPA Status Register</u> is Clear, this field has no effect on the Function substate.</p> <p>Default value is 0 0000b.</p>	RW

### 7.9.12.6 DPA Power Allocation Array



A-0764

Figure 7-237 DPA Power Allocation Array

Each Substate Power Allocation register indicates the power allocation value for its associated substate. The number of Substate Power Allocation registers implemented must be equal to the number of substates supported by Function, which is Substate\_Max plus one.

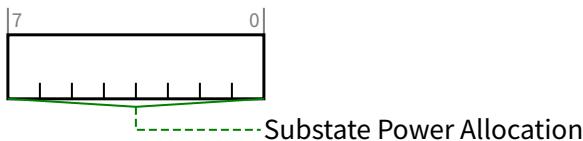


Figure 7-238 Substate Power Allocation Register (0 to Substate\_Max)

Table 7-193 Substate Power Allocation Register (0 to Substate\_Max)

Bit Location	Register Description	Attributes
7:0	<b>Substate Power Allocation</b> - The value in this field is multiplied by the <u>Power Allocation Scale</u> to determine power allocation in Watts for the associated substate.	<u>RO</u>

### 7.9.13 TPH Requester Extended Capability

The TPH Requester Extended Capability structure is required for all Functions that are capable of generating Request TLPs with TPH. For a Multi-Function Device, this capability must be present in each Function that is capable of generating Request TLPs with TPH.

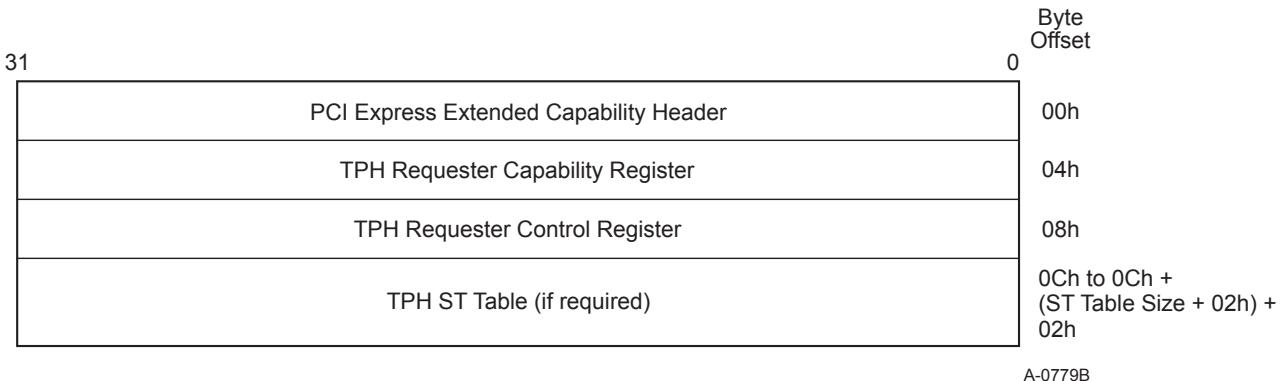


Figure 7-239 TPH Extended Capability Structure

#### 7.9.13.1 TPH Requester Extended Capability Header (Offset 00h)

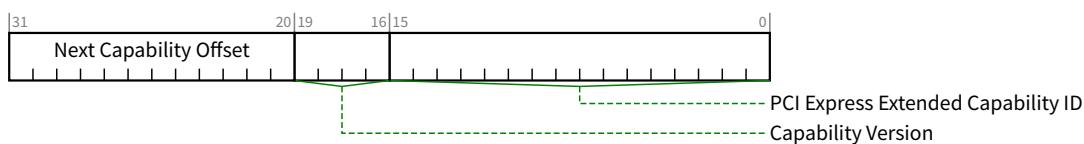


Figure 7-240 TPH Requester Extended Capability Header

Table 7-194 TPH Requester Extended Capability Header

Bit Location	Register Description	Attributes
15:0	<b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability.  PCI Express Extended Capability ID for the <u>TPH Requester Extended Capability</u> is 0017h.	<u>RO</u>
19:16	<b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the Capability structure present.  Must be 1h for this version of the specification.	<u>RO</u>
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Extended Capability structure or 000h if no other items exist in the linked list of Capabilities.	<u>RO</u>

### 7.9.13.2 TPH Requester Capability Register (Offset 04h)

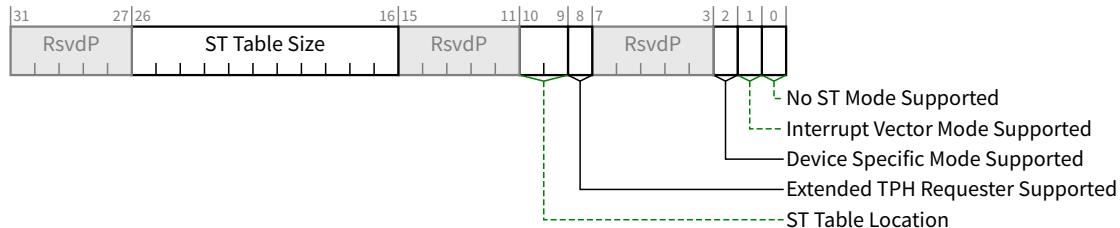


Figure 7-241 TPH Requester Capability Register

Table 7-195 TPH Requester Capability Register

Bit Location	Register Description	Attributes
0	<b>No ST Mode Supported</b> - If set indicates that the Function supports the <u>No ST Mode</u> of operation.  This mode is required to be supported by all Functions that implement this Capability structure. This bit must have a value of 1b.	<u>RO</u>
1	<b>Interrupt Vector Mode Supported</b> - If set indicates that the Function supports the Interrupt Vector Mode of operation.	<u>RO</u>
2	<b>Device Specific Mode Supported</b> - If set indicates that the Function supports the Device Specific Mode of operation.	<u>RO</u>
8	<b>Extended TPH Requester Supported</b> - If Set indicates that the Function is capable of generating Requests with a <u>TPH TLP Prefix</u> .  See <u>Section 2.2.7.1</u> for additional details.	<u>RO</u>
10:9	<b>ST Table Location</b> - Value indicates if and where the ST Table is located.  Defined Encodings are:  00b ST Table is not present 01b ST Table is located in the TPH Requester Extended Capability structure 10b ST Table is located in the MSI-X Table (see Section 7.7.2)	<u>RO</u>

Bit Location	Register Description	Attributes
	<p><b>11b</b> Reserved</p> <p>A Function that only supports the <u>No ST Mode</u> of operation must have a value of 00b in this field.</p> <p>A Function may report a value of 10b only if it implements an MSI-X Capability.</p>	
26:16	<p><b>ST Table Size</b> - Value indicates the maximum number of ST Table entries the Function may use. Software reads this field to determine the <u>ST Table Size N</u>, which is encoded as N-1. For example, a returned value of 000 0000 0011b indicates a table size of four entries.</p> <p>There is an upper limit of 64 entries when the ST Table is located in the <u>TPH Requester Extended Capability</u> structure.</p> <p>When the ST Table is located in the MSI-X Table, this value is limited by the size of the MSI-X Table.</p> <p>This field is only applicable for Functions that implement an ST Table as indicated by the <u>ST Table Location</u> field. Otherwise, the value in this field is undefined.</p>	<u>RO</u>

### 7.9.13.3 TPH Requester Control Register (Offset 08h)

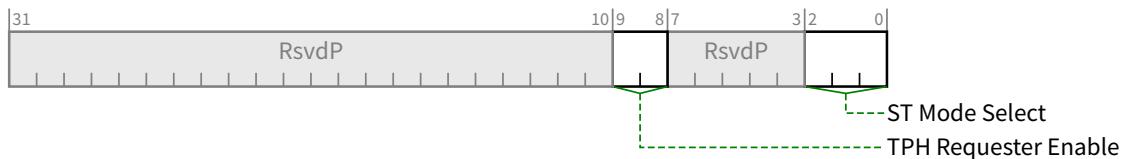


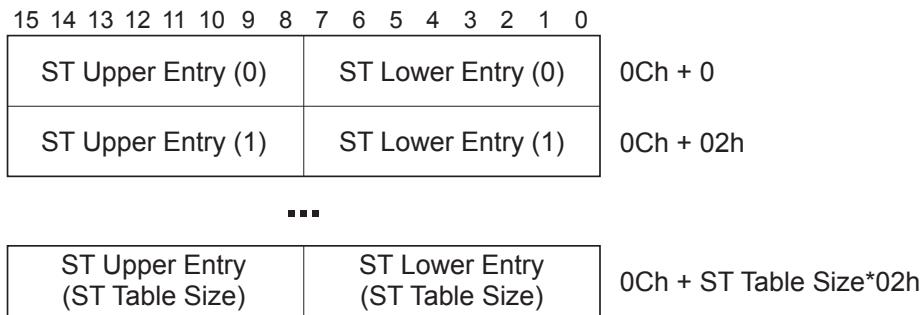
Figure 7-242 TPH Requester Control Register

Table 7-196 TPH Requester Control Register

Bit Location	Register Description	Attributes
2:0	<p><b>ST Mode Select</b> - selects the ST Mode of operation.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>000b</b> <b>No ST Mode</b></li> <li><b>001b</b> Interrupt Vector Mode</li> <li><b>010b</b> Device Specific Mode</li> <li><b>others</b> reserved for future use</li> </ul> <p>Functions that support only the <u>No ST Mode</u> of operation must hardwire this field to 000b.</p> <p>Function operation is undefined if software enables a mode of operation that does not correspond to a mode supported by the Function.</p> <p>The default value of this field is 000b.</p> <p>See <u>Section 6.17.3</u> for details on ST modes of operation.</p>	<u>RW</u>
9:8	<p><b>TPH Requester Enable</b> - Controls the ability to issue Request TLPs using either TPH or Extended TPH.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>00b</b> Function operating as a Requester is not permitted to issue Requests with TPH or Extended TPH</li> </ul>	<u>RW</u>

Bit Location	Register Description	Attributes
	<p><b>01b</b> Function operating as a Requester is permitted to issue Requests with TPH and is not permitted to issue Requests with Extended TPH</p> <p><b>10b</b> Reserved</p> <p><b>11b</b> Function operating as a Requester is permitted to issue Requests with TPH and Extended TPH</p> <p>Functions that advertise that they do not support Extended TPH are permitted to hardwire bit 9 of this field to 0b.</p> <p>The default value of this field is 00b.</p>	

#### 7.9.13.4 TPH ST Table (Starting from Offset 0Ch)



A-0783B

Figure 7-243 TPH ST Table

The TPH ST Table must be implemented in the TPH Requester Extended Capability structure if the value of the ST Table Location field is 01b. For all other values, the ST Entry registers must not be implemented. Each implemented ST Entry is 16 bits. The number of ST Entry registers implemented must be equal to the number of ST Table entries supported by the Function, which is the value of the ST Table Size field plus one.

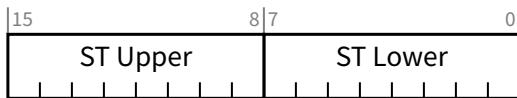


Figure 7-244 TPH ST Table Entry

Table 7-197 TPH ST Table Entry

Bit Location	Register Description	Attributes
7:0	<p><b>ST Lower</b> - This field contains the lower 8 bits of a Steering Tag.</p> <p>Default value of this field is 00h.</p>	RW

Bit Location	Register Description	Attributes
15:8	<p><b>ST Upper</b> - If the Function's Extended TPH Requester Supported bit is Set, then this field contains the upper 8 bits of a Steering Tag. Otherwise, this field is <u>RsvdP</u>.</p> <p>Default value of this field is 00h.</p>	<u>RW</u>

## 7.9.14 LN Requester Extended Capability (LNR Capability)

The LN Requester Extended Capability is an optional normative capability for Endpoints. All Endpoints that support LN protocol as a Requester must implement this capability. See [Section 6.21](#). This capability may be implemented by any type of Endpoint, but not by any other Function type.

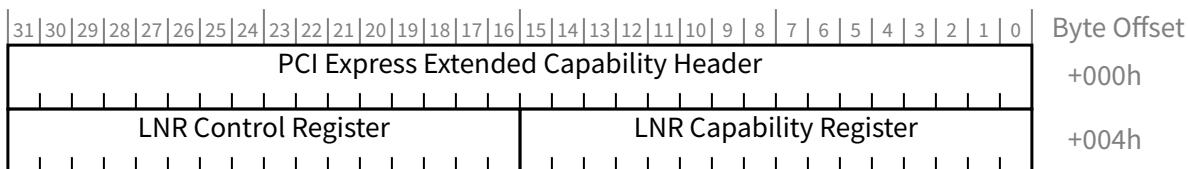


Figure 7-245 LN Requester Extended Capability

### 7.9.14.1 LNR Extended Capability Header (Offset 00h)



Figure 7-246 LNR Extended Capability Header

Table 7-198 LNR Extended Capability Header

Bit Location	Register Description	Attributes
15:0	<p><b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the extended capability.</p> <p>PCI Express Extended Capability ID for the LNR Extended Capability is 001Ch.</p>	<u>RO</u>
19:16	<p><b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the capability structure present.</p> <p>Must be 1h for this version of the specification.</p>	<u>RO</u>
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Extended Capability structure or 000h if no other items exist in the linked list of capabilities.	<u>RO</u>

### 7.9.14.2 LNR Capability Register (Offset 04h)

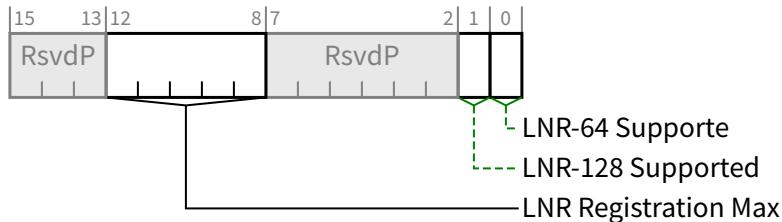


Figure 7-247 LNR Capability Register

Table 7-199 LNR Capability Register

Bit Location	Register Description	Attributes
0	<b>LNR-64 Supported</b> - This bit must be 1b if the Endpoint supports LN protocol for 64-byte cachelines as a Requester; otherwise, must be 0b. See <a href="#">Section 6.21.4</a> for additional details.	RO
1	<b>LNR-128 Supported</b> - This bit must be 1b if the Endpoint supports LN protocol for 128-byte cachelines as a Requester; otherwise, must be 0b.	RO
12:8	<b>LNR Registration Max</b> - This field, encoded as a power of 2, indicates the maximum number of cachelines that this LN Requester is capable of registering concurrently. For example, a value of 00101b indicates that the LN Requester might be capable of registering up to 32 cachelines ( $2^5$ ) concurrently, and is capable of registering greater than 16.	RO

### 7.9.14.3 LNR Control Register (Offset 06h)

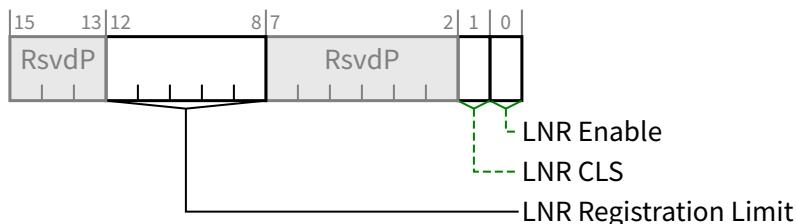


Figure 7-248 LNR Control Register

Table 7-200 LNR Control Register

Bit Location	Register Description	Attributes
0	<b>LNR Enable</b> - When this bit is Set, the Endpoint is enabled to operate as an LN Requester. Software is permitted to Clear this bit at any time. See <a href="#">Section 6.21.4</a> for requirements regarding the LNR's internal registration state.	RW

Bit Location	Register Description	Attributes
	Default value of this bit is 0b.	
1	<p><b>LNR CLS</b> - This bit controls or indicates the cache line size used with LN protocol by this Requester. See <a href="#">Section 6.21.4</a> for restrictions on setting and modifying this bit.</p> <p>If this bit is Clear, the cache line size is 64 bytes. If this bit is Set, the cache line size is 128 bytes.</p> <p>If this LN Requester supports only one cache line size, this bit is permitted to be hardwired to indicate that size. Otherwise, the default value of this bit is 0b.</p>	<u>RW</u>
12:8	<p><b>LNR Registration Limit</b> - This field, encoded as a power of 2, imposes a limit on the number of cachelines that this LN Requester is permitted to register concurrently. For example, a value of 00100b indicates that the LN Requester must not register more than 16 cachelines (<math>2^4</math>) concurrently. See <a href="#">Section 6.21.4</a> for restrictions on modifying this field.</p> <p>The default value of this field is 11111b.</p>	<u>RW</u>

## 7.9.15 DPC Extended Capability

The Downstream Port Containment (DPC) Extended Capability is an optional normative capability that provides a mechanism for Downstream Ports to contain uncorrectable errors and enable software to recover from them. See [Section 6.2.10](#). This capability may be implemented by a Root Port or a Switch Downstream Port. It is not applicable to any other Device/Port type.

If a Downstream Port implements the DPC Extended Capability, that Port must also be capable of reporting the DL\_Active state, and indicate so by Setting the Data Link Layer Link Active Reporting Capable bit in the Link Capabilities Register. See [Section 7.5.3.6](#).

If a Downstream Port implements the DPC Extended Capability, it is strongly recommended for that Port to support ERR\_COR Subclass capability, and indicate so by Setting the ERR\_COR Subclass Capable bit in the Device Capabilities Register. See [Section 7.5.3.3](#).

The various RP PIO registers must be implemented only by Root Ports that support RP Extensions for DPC, as indicated by the RP Extensions for DPC bit in the DPC Capability Register.

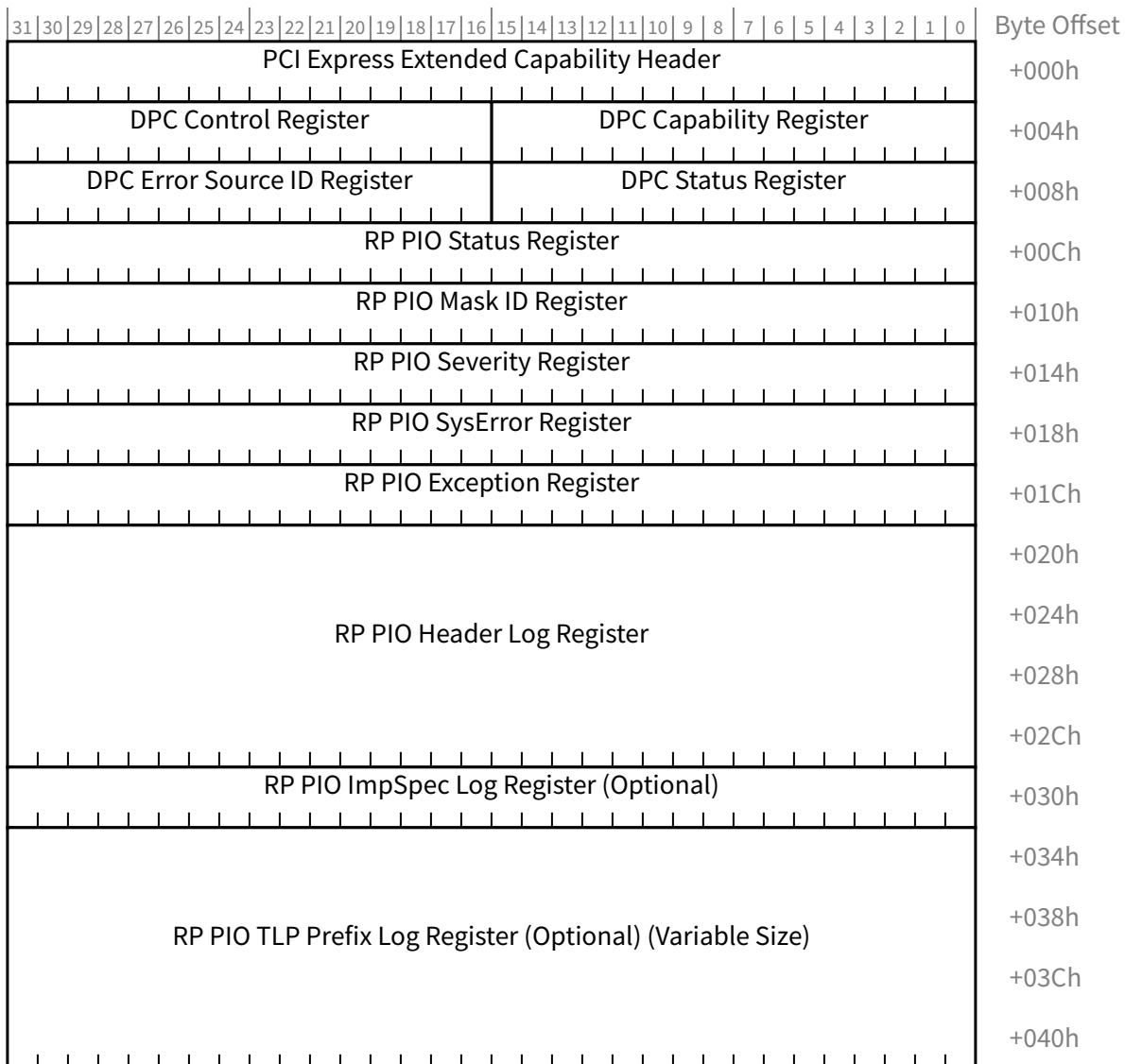


Figure 7-249 DPC Extended Capability

### 7.9.15.1 DPC Extended Capability Header (Offset 00h)

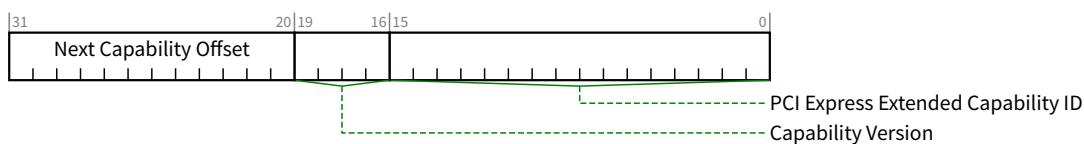


Figure 7-250 DPC Extended Capability Header

Table 7-201 DPC Extended Capability Header

Bit Location	Register Description	Attributes
15:0	<b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the extended capability.  PCI Express Extended Capability ID for the DPC Extended Capability is 001Dh.	<u>RO</u>
19:16	<b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the capability structure present.  Must be 1h for this version of the specification.	<u>RO</u>
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Extended Capability structure or 000h if no other items exist in the linked list of capabilities.	<u>RO</u>

### 7.9.15.2 DPC Capability Register (Offset 04h)

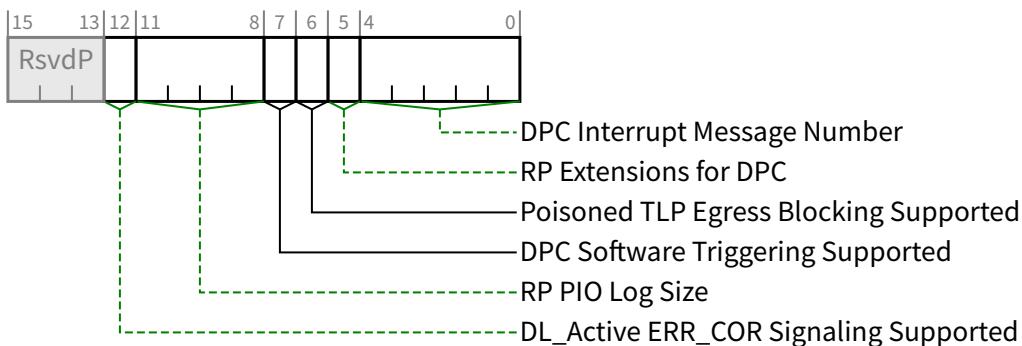


Figure 7-251 DPC Capability Register

Table 7-202 DPC Capability Register

Bit Location	Register Description	Attributes
4:0	<b>DPC Interrupt Message Number</b> - This field indicates which MSI/MSI-X vector is used for the interrupt message generated in association with the DPC Capability structure.  For MSI, the value in this field indicates the offset between the base Message Data and the interrupt message that is generated. Hardware is required to update this field so that it is correct if the number of MSI Messages assigned to the Function changes when software writes to the <u>Multiple Message Enable</u> field in the <u>Message Control Register for MSI</u> .  For MSI-X, the value in this field indicates which MSI-X Table entry is used to generate the interrupt message. The entry must be one of the first 32 entries even if the Function implements more than 32 entries. For a given MSI-X implementation, the entry must remain constant.  If both MSI and MSI-X are implemented, they are permitted to use different vectors, though software is permitted to enable only one mechanism at a time. If MSI-X is enabled, the value in this field must indicate the vector for MSI-X. If MSI is enabled or neither is enabled, the value in this field must indicate the vector for MSI. If software enables both MSI and MSI-X at the same time, the value in this field is undefined.	<u>RO</u>

Bit Location	Register Description	Attributes
5	<b>RP Extensions for DPC</b> - If Set, this bit indicates that a Root Port supports a defined set of DPC Extensions that are specific to Root Ports. Switch Downstream Ports must not Set this bit.	RO
6	<b>Poisoned TLP Egress Blocking Supported</b> - If Set, this bit indicates that the Root Port or Switch Downstream Port supports the ability to block the transmission of a poisoned TLP from its Egress Port. Root Ports that support <u>RP Extensions for DPC</u> must Set this bit.	RO
7	<b>DPC Software Triggering Supported</b> - If Set, this bit indicates that a Root Port or Switch Downstream Port supports the ability for software to trigger DPC. Root Ports that support <u>RP Extensions for DPC</u> must Set this bit.	RO
11:8	<b>RP PIO Log Size</b> - This field indicates how many DWORDs are allocated for the RP PIO log registers, comprised by the RP PIO Header Log, the RP PIO ImpSpec Log, and RP PIO TLP Prefix Log. If the Root Port supports <u>RP Extensions for DPC</u> , the value of this field must be 4 or greater; otherwise, the value of this field must be 0. See <a href="#">Section 7.9.15.11</a> , <a href="#">Section 7.9.15.12</a> , and <a href="#">Section 7.9.15.13</a> .	RO
12	<b>DL_Active ERR_COR Signaling Supported</b> - If Set, this bit indicates that the Root Port or Switch Downstream Port supports the ability to signal with ERR_COR when the Link transitions to the DL_Active state. Root Ports that support <u>RP Extensions for DPC</u> must Set this bit.	RO

### 7.9.15.3 DPC Control Register (Offset 06h)

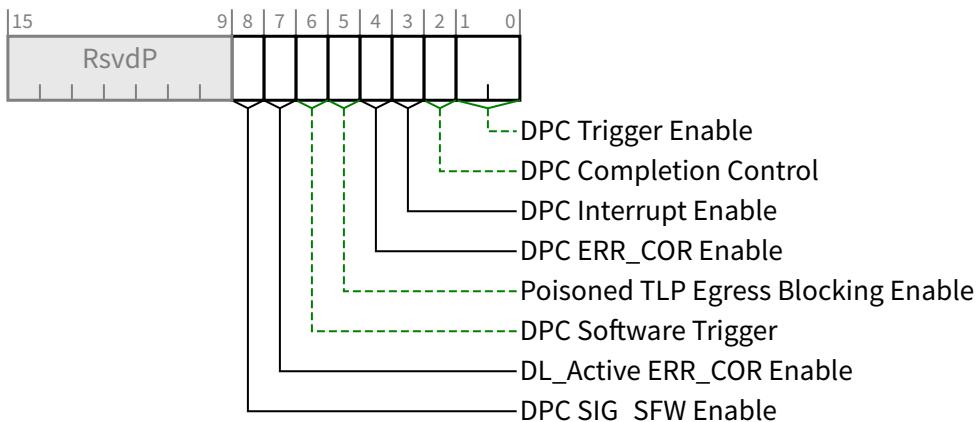


Figure 7-252 DPC Control Register

Table 7-203 DPC Control Register

Bit Location	Register Description	Attributes
1:0	<b>DPC Trigger Enable</b> - This field enables DPC and controls the conditions that cause DPC to be triggered. Defined encodings are: <b>00b</b> DPC is disabled <b>01b</b> DPC is enabled and is triggered when the Downstream Port detects an unmasked uncorrectable error or when the Downstream Port receives an ERR_FATAL Message	RW

Bit Location	Register Description	Attributes
	<p><b>10b</b> DPC is enabled and is triggered when the Downstream Port detects an unmasked uncorrectable error or when the Downstream Port receives an <u>ERR_NONFATAL</u> or <u>ERR_FATAL</u> Message</p> <p><b>11b</b> Reserved</p> <p>Default value of this field is 00b.</p>	
2	<p><b>DPC Completion Control</b> - This bit controls the Completion Status for Completions formed during DPC. See <u>Section 2.9.3</u>.</p> <p>Defined encodings are:</p> <p><b>0b</b> Completer Abort (CA) Completion Status</p> <p><b>1b</b> Unsupported Request (UR) Completion Status</p> <p>Default value of this bit is 0b.</p>	<u>RW</u>
3	<p><b>DPC Interrupt Enable</b> - When Set, this bit enables the generation of an interrupt to indicate that DPC has been triggered. See <u>Section 6.2.10.1</u>.</p> <p>Default value of this bit is 0b.</p>	<u>RW</u>
4	<p><b>DPC ERR_COR Enable</b> - When Set, this bit enables the sending of an <u>ERR_COR</u> Message to indicate that DPC has been triggered. See <u>Section 6.2.10.2</u>.</p> <p>Default value of this bit is 0b.</p>	<u>RW</u>
5	<p><b>Poisoned TLP Egress Blocking Enable</b> - This bit must be <u>RW</u> if the <u>Poisoned TLP Egress Blocking Supported</u> bit is Set; otherwise, it is permitted to be hardwired to 0b. Software must not Set this bit unless the <u>Poisoned TLP Egress Blocking Supported</u> bit is Set.</p> <p>When Set, this bit enables the associated Egress Port to block the transmission of poisoned TLPs. See <u>Section 2.7.2.2</u>.</p> <p>Default value of this bit is 0b.</p>	<u>RW/RO</u>
6	<p><b>DPC Software Trigger</b> - This bit must be <u>RW</u> if the <u>DPC Software Triggering Supported</u> bit is Set; otherwise, it is permitted to be hardwired to 0b.</p> <p>If DPC is enabled and the <u>DPC Trigger Status</u> bit is Clear, when software writes 1b to this bit, DPC is triggered. Otherwise, software writing a 1b to this bit has no effect.</p> <p>It is permitted to write 1b to this bit while simultaneously writing updated values to other fields in this register, notably the <u>DPC Trigger Enable</u> field. For this case, the <u>DPC Software Trigger</u> semantics are based on the updated value of the <u>DPC Trigger Enable</u> field.</p> <p>This bit always returns 0b when read.</p>	<u>RW/RO</u>
7	<p><b>DL_Active ERR_COR Enable</b> - This bit must be <u>RW</u> if the <u>DL_Active ERR_COR Signaling Supported</u> bit is Set; otherwise, it is permitted to be hardwired to 0b. Software must not Set this bit unless the <u>DL_Active ERR_COR Signaling Supported</u> bit is Set.</p> <p>When Set, this bit enables the associated Downstream Port to signal with <u>ERR_COR</u> when the Link transitions to the <u>DL_Active</u> state. See <u>Section 6.2.10.5</u>.</p> <p>Default value of this bit is 0b.</p>	<u>RW/RO</u>
8	<p><b>DPC SIG_SFW Enable</b> - This bit must be implemented if the <u>ERR_COR Subclass Capable</u> bit in the <u>Device Capabilities Register</u> is Set; otherwise, it is permitted to be hardwired to 0b. If the <u>ERR_COR Subclass Capable</u> bit is Clear and software Sets this bit, the behavior is undefined.</p> <p>When Set, this bit enables sending an <u>ERR_COR</u> Message to indicate a DPC event that's been enabled for <u>ERR_COR</u> signaling. See <u>Section 6.2.10.2</u> and <u>Section 6.2.10.5</u>. This is an additional and alternative way to enable overall DPC <u>ERR_COR</u> signaling beyond the <u>Correctable Error Reporting Enable</u> bit in the</p>	<u>RW/RO</u>

Bit Location	Register Description	Attributes
	<p>Device Control Register. This bit does not affect a Function's ability to send <u>ERR_COR</u> Messages other than the ECS SIG_SFW subclass.</p> <p>Default value of this bit is 0b.</p>	

### 7.9.15.4 DPC Status Register (Offset 08h)

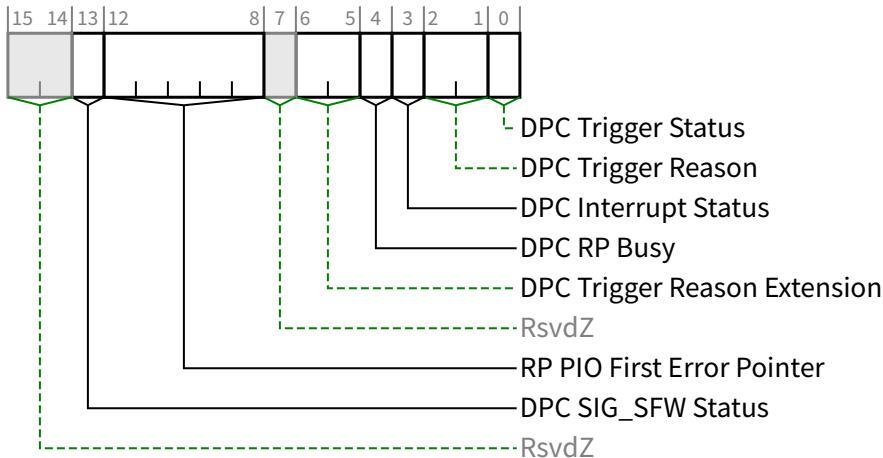


Figure 7-253 DPC Status Register

Table 7-204 DPC Status Register

Bit Location	Register Description	Attributes
0	<p><b>DPC Trigger Status</b> - When Set, this bit indicates that DPC has been triggered, and by definition the Port is “in DPC”. DPC is event triggered.</p> <p>While this bit is Set, hardware must direct the LTSSM to the Disabled State. This bit must be cleared before the LTSSM can be released from the Disabled State, after which the Port is no longer in DPC, and the LTSSM must transition to the Detect State. See <u>Section 6.2.10</u> for requirements on how long software must leave the Downstream Port in DPC. Once these requirements are met, software is permitted to clear this bit regardless of the state of other status bits associated with the triggering event.</p> <p>After clearing this bit, software must honor timing requirements defined in <u>Section 6.6.1</u> with respect to the first Configuration Read following a Conventional Reset.</p> <p>Default value of this bit is 0b.</p>	<u>RW1CS</u>
2:1	<p><b>DPC Trigger Reason</b> - This field indicates why DPC has been triggered. Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>00b</b> DPC was triggered due to an unmasked uncorrectable error</li> <li><b>01b</b> DPC was triggered due to receiving an ERR_NONFATAL</li> <li><b>10b</b> DPC was triggered due to receiving an ERR_FATAL</li> <li><b>11b</b> DPC was triggered due to a reason that is indicated by the <u>DPC Trigger Reason Extension</u> field.</li> </ul>	<u>ROS</u>

Bit Location	Register Description	Attributes
	This field is valid only when the <u>DPC Trigger Status</u> bit is Set; otherwise the value of this field is undefined.	
3	<p><b>DPC Interrupt Status</b> - This bit is Set if DPC is triggered while the <u>DPC Interrupt Enable</u> bit is Set. This may cause the generation of an interrupt. See <u>Section 6.2.10.1</u>.</p> <p>Default value of this bit is 0b.</p>	<u>RW1CS</u>
4	<p><b>DPC RP Busy</b> - When the <u>DPC Trigger Status</u> bit is Set and this bit is Set, the Root Port is busy with internal activity that must complete before software is permitted to Clear the <u>DPC Trigger Status</u> bit. If software Clears the <u>DPC Trigger Status</u> bit while this bit is Set, the behavior is undefined.</p> <p>This field is valid only when the <u>DPC Trigger Status</u> bit is Set; otherwise the value of this field is undefined.</p> <p>This bit is applicable only for Root Ports that support <u>RP Extensions for DPC</u>, and is Reserved for Switch Downstream Ports.</p> <p>Default value of this bit is undefined.</p>	<u>RO/RsvdZ</u>
6:5	<p><b>DPC Trigger Reason Extension</b> - This field serves as an extension to the <u>DPC Trigger Reason</u> field. When that field is valid and has a value of 11b, this field indicates why DPC has been triggered. Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>00b</b> DPC was triggered due to an RP PIO error</li> <li><b>01b</b> DPC was triggered due to the DPC Software Trigger bit</li> <li><b>10b</b> Reserved</li> <li><b>11b</b> Reserved</li> </ul> <p>This field is valid only when the <u>DPC Trigger Status</u> bit is Set and the value of the <u>DPC Trigger Reason</u> field is 11b; otherwise the value of this field is undefined.</p>	<u>ROS</u>
12:8	<p><b>RP PIO First Error Pointer</b> - The value of this field identifies a bit position in the <u>RP PIO Status</u> Register, and this field is considered valid when that bit is Set. When this field is valid, and software writes a 1b to the indicated RP PIO Status bit (thus clearing it), this field must revert to its default value.</p> <p>This field is applicable only for Root Ports that support <u>RP Extensions for DPC</u>, and otherwise is Reserved.</p> <p>If this field is not Reserved, its default value is 11111b, indicating a permanently Reserved RP PIO Status bit, thus guaranteeing that this field is not considered valid.</p>	<u>ROS/RsvdZ</u>
13	<p><b>DPC SIG_SFW Status</b> - If the Function supports ERR_COR Subclass capability, this bit must be implemented; otherwise, it must be hardwired to 0b. If implemented, this bit is Set when a SIG_SFW ERR_COR Message is sent to signal a DPC event. See <u>Section 6.2.10.2</u> and <u>Section 6.2.10.5</u>.</p> <p>Default value of this bit is 0b</p>	<u>RW1CS/RsvdZ</u>

### 7.9.15.5 DPC Error Source ID Register (Offset 0Ah)

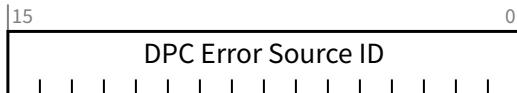


Figure 7-254 DPC Error Source ID Register

Table 7-205 DPC Error Source ID Register

Bit Location	Register Description	Attributes
15:0	<b>DPC Error Source ID</b> - When the DPC Trigger Reason field indicates that DPC was triggered due to the reception of an <u>ERR_NONFATAL</u> or <u>ERR_FATAL</u> , this register contains the Requester ID of the received Message. Otherwise, the value of this register is undefined.	<u>ROS</u>

### 7.9.15.6 RP PIO Status Register (Offset 0Ch)

This register is present only in Root Ports that support RP Extensions for DPC. See [Section 6.2.10.3](#).

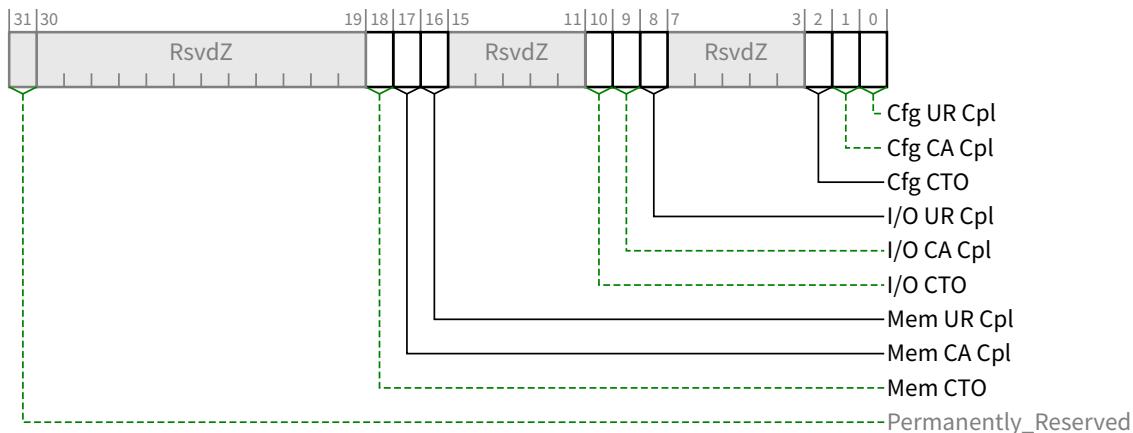


Figure 7-255 RP PIO Status Register

Table 7-206 RP PIO Status Register

Bit Location	Register Description	Attributes	Default
0	<b>Cfg UR Cpl</b> - Configuration Request received UR Completion	<u>RW1CS</u>	0b
1	<b>Cfg CA Cpl</b> - Configuration Request received CA Completion	<u>RW1CS</u>	0b
2	<b>Cfg CTO</b> - Configuration Request Completion Timeout	<u>RW1CS</u>	0b

Bit Location	Register Description	Attributes	Default
8	<b>I/O UR Cpl</b> - I/O Request received UR Completion	RW1CS	0b
9	<b>I/O CA Cpl</b> - I/O Request received CA Completion	RW1CS	0b
10	<b>I/O CTO</b> - I/O Request Completion Timeout	RW1CS	0b
16	<b>Mem UR Cpl</b> - Memory Request received UR Completion	RW1CS	0b
17	<b>Mem CA Cpl</b> - Memory Request received CA Completion	RW1CS	0b
18	<b>Mem CTO</b> - Memory Request Completion Timeout	RW1CS	0b
31	<b>Permanently Reserved</b> , since the default RP PIO First Error Pointer field value points to it.	RsvdZ	0b

### 7.9.15.7 RP PIO Mask Register (Offset 10h)

This register is present only in Root Ports that support RP Extensions for DPC. See [Section 6.2.10.3](#).

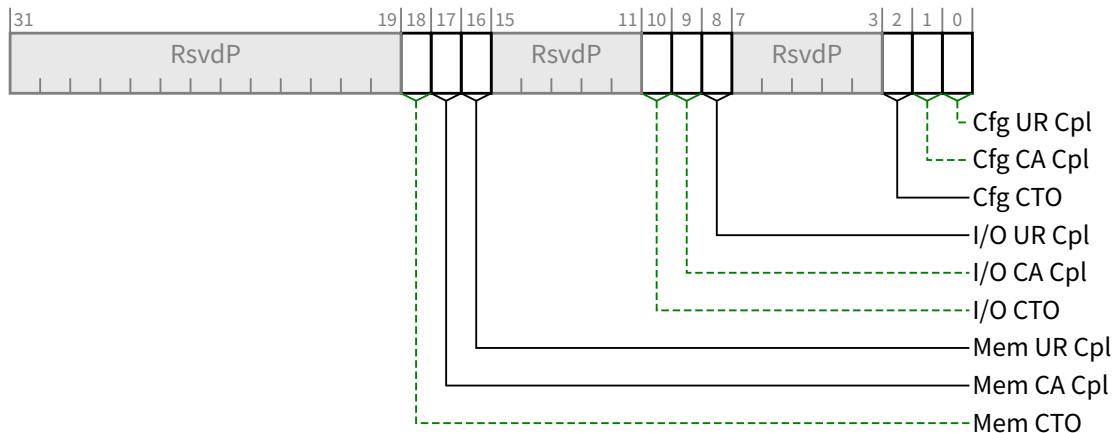


Figure 7-256 RP PIO Mask Register

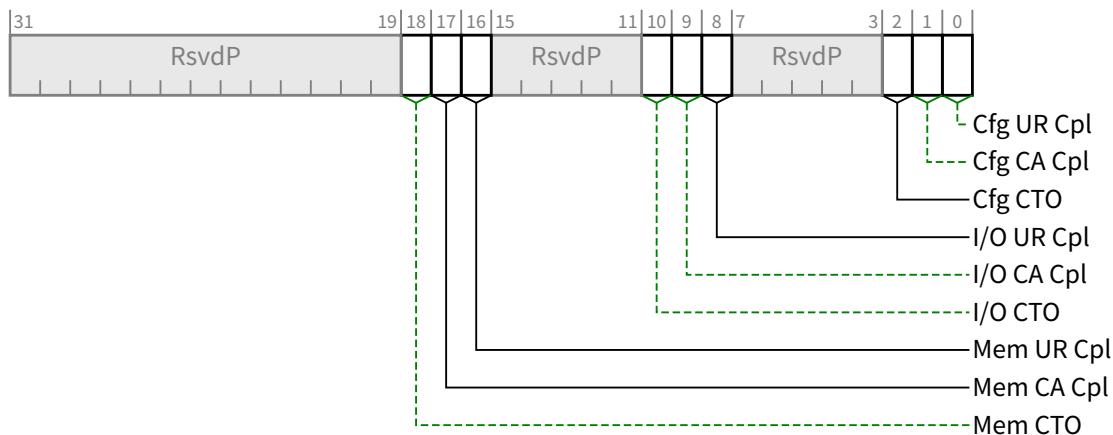
Table 7-207 RP PIO Mask Register

Bit Location	Register Description	Attributes	Default
0	<b>Cfg UR Cpl</b> - Configuration Request received UR Completion	RWS	1b
1	<b>Cfg CA Cpl</b> - Configuration Request received CA Completion	RWS	1b
2	<b>Cfg CTO</b> - Configuration Request Completion Timeout	RWS	1b
8	<b>I/O UR Cpl</b> - I/O Request received UR Completion	RWS	1b
9	<b>I/O CA Cpl</b> - I/O Request received CA Completion	RWS	1b
10	<b>I/O CTO</b> - I/O Request Completion Timeout	RWS	1b

Bit Location	Register Description	Attributes	Default
16	<b>Mem UR Cpl</b> - Memory Request received UR Completion	RWS	1b
17	<b>Mem CA Cpl</b> - Memory Request received CA Completion	RWS	1b
18	<b>Mem CTO</b> - Memory Request Completion Timeout	RWS	1b

#### **7.9.15.8 RP PIO Severity Register (Offset 14h)**

This register is present only in Root Ports that support RP Extensions for DPC. See Section 6.2.10.3 .



*Figure 7-257 RP PIO Severity Register*

*Table 7-208 RP PIO Severity Register*

Bit Location	Register Description	Attributes	Default
0	<b><i>Cfg UR Cpl</i></b> - Configuration Request received UR Completion	RWS	0b
1	<b><i>Cfg CA Cpl</i></b> - Configuration Request received CA Completion	RWS	0b
2	<b><i>Cfg CTO</i></b> - Configuration Request Completion Timeout	RWS	0b
8	<b><i>I/O UR Cpl</i></b> - I/O Request received UR Completion	RWS	0b
9	<b><i>I/O CA Cpl</i></b> - I/O Request received CA Completion	RWS	0b
10	<b><i>I/O CTO</i></b> - I/O Request Completion Timeout	RWS	0b
16	<b><i>Mem UR Cpl</i></b> - Memory Request received UR Completion	RWS	0b
17	<b><i>Mem CA Cpl</i></b> - Memory Request received CA Completion	RWS	0b
18	<b><i>Mem CTO</i></b> - Memory Request Completion Timeout	RWS	0b

### 7.9.15.9 RP PIO SysError Register (Offset 18h)

This register is present only in Root Ports that support RP Extensions for DPC. See Section 6.2.10.3 .

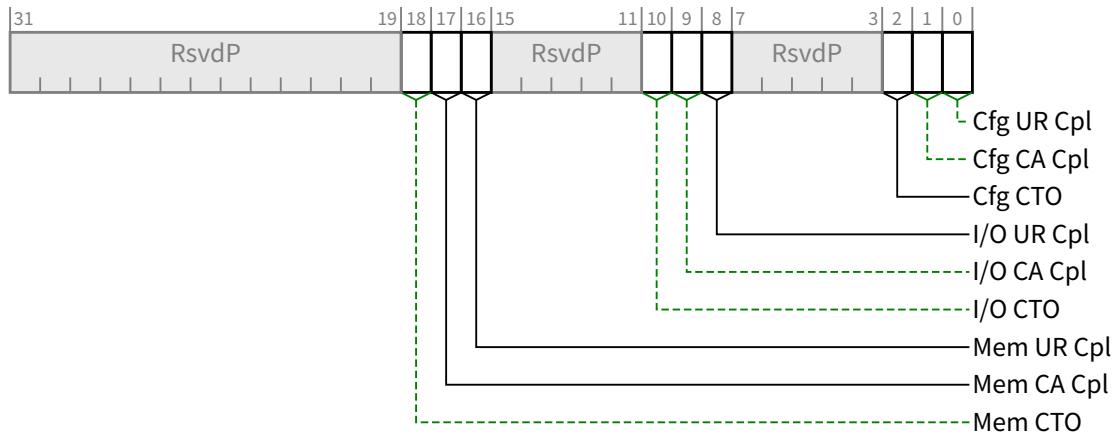


Figure 7-258 RP PIO SysError Register

Table 7-209 RP PIO SysError Register

Bit Location	Register Description	Attributes	Default
0	<b>Cfg UR Cpl</b> - Configuration Request received UR Completion	RWS	0b
1	<b>Cfg CA Cpl</b> - Configuration Request received CA Completion	RWS	0b
2	<b>Cfg CTO</b> - Configuration Request Completion Timeout	RWS	0b
8	<b>I/O UR Cpl</b> - I/O Request received UR Completion	RWS	0b
9	<b>I/O CA Cpl</b> - I/O Request received CA Completion	RWS	0b
10	<b>I/O CTO</b> - I/O Request Completion Timeout	RWS	0b
16	<b>Mem UR Cpl</b> - Memory Request received UR Completion	RWS	0b
17	<b>Mem CA Cpl</b> - Memory Request received CA Completion	RWS	0b
18	<b>Mem CTO</b> - Memory Request Completion Timeout	RWS	0b

### 7.9.15.10 RP PIO Exception Register (Offset 1Ch)

This register is present only in Root Ports that support RP Extensions for DPC. See Section 6.2.10.3 .

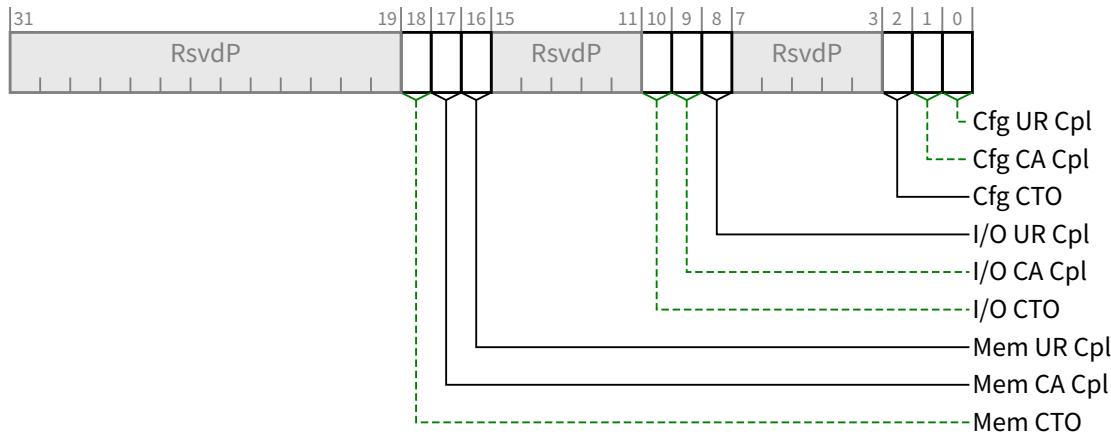


Figure 7-259 RP PIO Exception Register

Table 7-210 RP PIO Exception Register

Bit Location	Register Description	Attributes	Default
0	<b>Cfg UR Cpl</b> - Configuration Request received UR Completion	RWS	0b
1	<b>Cfg CA Cpl</b> - Configuration Request received CA Completion	RWS	0b
2	<b>Cfg CTO</b> - Configuration Request Completion Timeout	RWS	0b
8	<b>I/O UR Cpl</b> - I/O Request received UR Completion	RWS	0b
9	<b>I/O CA Cpl</b> - I/O Request received CA Completion	RWS	0b
10	<b>I/O CTO</b> - I/O Request Completion Timeout	RWS	0b
16	<b>Mem UR Cpl</b> - Memory Request received UR Completion	RWS	0b
17	<b>Mem CA Cpl</b> - Memory Request received CA Completion	RWS	0b
18	<b>Mem CTO</b> - Memory Request Completion Timeout	RWS	0b

### 7.9.15.11 RP PIO Header Log Register (Offset 20h)

This register is implemented only in Root Ports that support RP Extensions for DPC. The RP PIO Header Log Register contains the header from the Request TLP associated with a recorded RP PIO error. Refer to Section 6.2.10.3 for further details. This register is 16 bytes and is formatted identically to the Header Log register in AER. See Section 7.8.4.8.

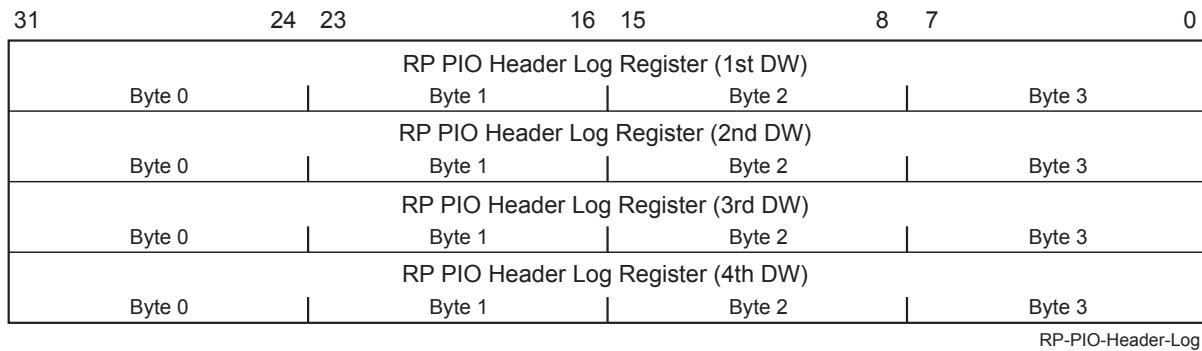


Figure 7-260 RP PIO Header Log Register

Table 7-211 RP PIO Header Log Register

Bit Location	Register Description	Attributes	Default
127:0	<b>TLP Header</b> - of the TLP associated with the error	ROS	0

### 7.9.15.12 RP PIO ImpSpec Log Register (Offset 30h)

This register is permitted to be implemented only in Root Ports that support RP Extensions for DPC. The RP PIO ImpSpec Log Register, if implemented, contains implementation-specific information associated with the recorded error, e.g., indicating the source of the Request TLP. Space is allocated for this register if the value of the RP PIO Log Size field is 5 or greater. If space is allocated for the register, but the register is not implemented, the bits must be hardwired to 0b.

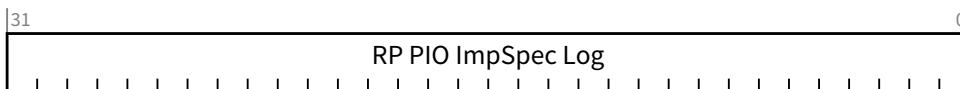


Figure 7-261 RP PIO ImpSpec Log Register

Table 7-212 RP PIO ImpSpec Log Register

Bit Location	Register Description	Attributes	Default
31:0	<b>RP PIO ImpSpec Log</b>	ROS	0

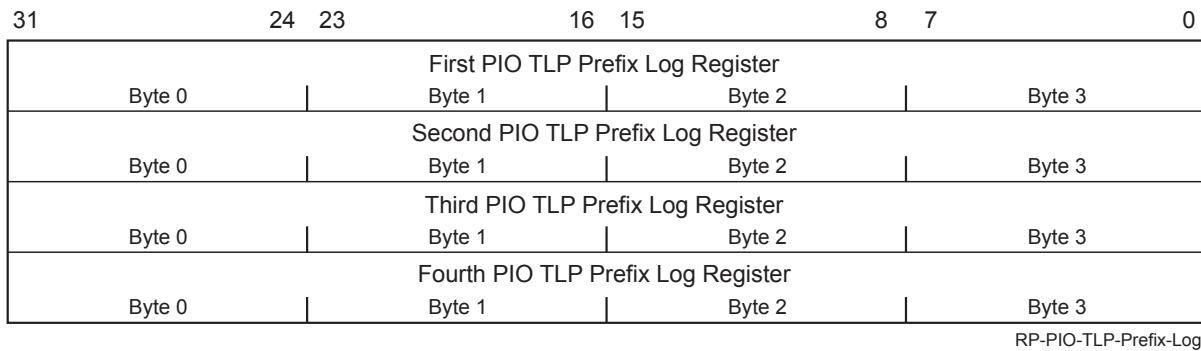
### 7.9.15.13 RP PIO TLP Prefix Log Register (Offset 34h)

This register is permitted to be implemented only in Root Ports that support RP Extensions for DPC. The RP PIO TLP Prefix Log Register contains any End-End TLP Prefixes from the TLP corresponding to a recorded RP PIO error. Refer to Section 6.2.10.3 for further details.

If the Root Port supports tracking Non-Posted Requests that contain End-End TLP Prefixes, this register must be implemented, and must be of sufficient size to record the maximum number of End-End TLP Prefixes for any tracked Request. See Section 2.9.3. The allocated size in DWORDs of the RP PIO TLP Prefix Log Register is the RP PIO Log Size.

minus 5 if the RP PIO Log Size is 9 or less, or 4 if the RP PIO Log Size is greater than 9. The implemented size of the TLP Prefix Log must be less than or equal to the Root Port's Max End-End TLP Prefixes field value. For the case where the Root Port never transmits Non-Posted Requests containing End-End TLP Prefixes, the allocated and implemented size of the TLP Prefix Log is permitted to be 0. Any DWORDs allocated but not implemented must be hardwired to zero.

This register is formatted identically to the TLP Prefix Log register in AER, although this register's allocated size is variable, whereas the register in AER is always 4 DWORDs. See [Section 7.8.4.12](#). The First TLP Prefix Log register contains the first End-End TLP Prefix from the TLP, the Second TLP Prefix Log register contains the second End-End TLP Prefix, and so forth. If the TLP contains fewer TLP Prefixes than this register accommodates, any remaining TLP Prefix Log registers must contain zero.



*Figure 7-262 RP PIO TLP Prefix Log Register*

*Table 7-213 RP PIO TLP Prefix Log Register*

Bit Location	Register Description	Attributes	Default
127:0	<b>RP PIO TLP Prefix Log</b>	ROS	0

## 7.9.16 Precision Time Management Extended Capability (PTM Capability)

The Precision Time Management Extended Capability is an optional Extended Capability for discovering and controlling the distribution of a PTM Hierarchy. For Root Complexes, this Capability is required in any Root Port, RCiEP, or RCRB that supports PTM. For Endpoints and Switch Upstream Ports that support PTM, this Capability is required in exactly one Function of the Upstream Port and that Capability controls the PTM behavior of all PTM capable Functions associated with that Upstream Port. For Switch Downstream Ports, PTM behavior is controlled by the same PTM Capability that controls the associated Switch Upstream Port. The PTM Capability is not permitted in Bridges, Switch Downstream Ports, and Root Complex Event Collectors.

For Switches, a single instance of this Capability controls behavior for the entire Switch. If the Upstream Port of the Switch is associated with an MFD, it is not required that the controlling Function be the Function corresponding to the Switch Upstream Port. For a given Switch, if this Capability is present, all Downstream Ports of the Switch must implement the requirements defined in [Section 6.22.3.2](#).

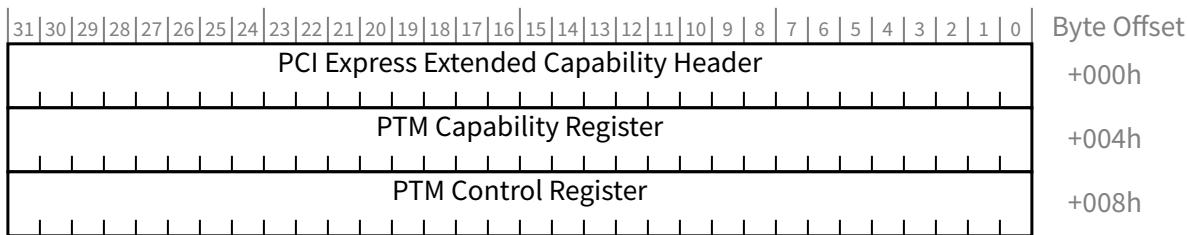


Figure 7-263 PTM Capability Structure

### 7.9.16.1 PTM Extended Capability Header (Offset 00h)

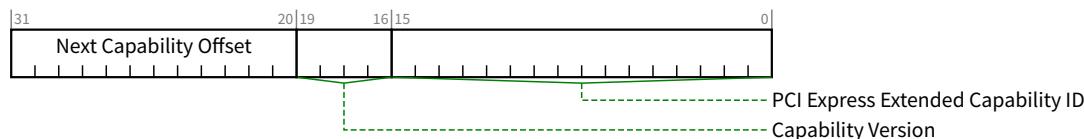


Figure 7-264 PTM Extended Capability Header

Table 7-214 PTM Extended Capability Header

Bit Location	Register Description	Attributes
15:0	<b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability. PCI Express Extended Capability ID for the Precision Time Measurement Capability is 001Fh.	RO
19:16	<b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the Capability structure present. Must be 1h for this version of the specification.	RO
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Extended Capability structure or 000h if no other items exist in the linked list of Capabilities.	RO

### 7.9.16.2 PTM Capability Register (Offset 04h)

This register describes a Function's support for Precision Time Measurement. Not all fields within this register apply to all Functions capable of implementing PTM.

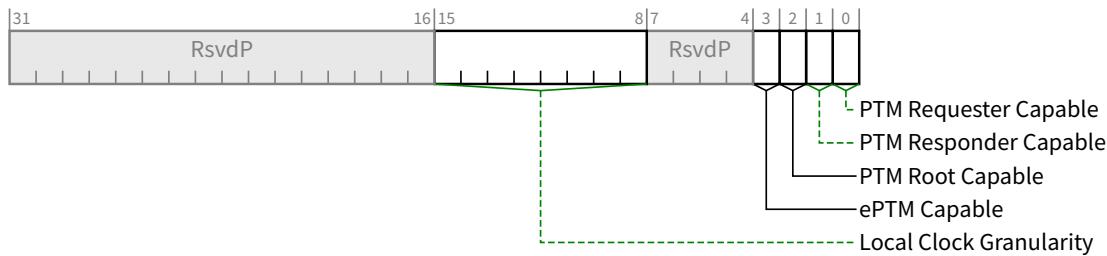


Figure 7-265 PTM Capability Register

Table 7-215 PTM Capability Register

Bit Location	Register Description	Attributes						
0	<p><b>PTM Requester Capable</b> - Indicates the Function implements the PTM Requester role (see <a href="#">Section 6.22.3.1</a>).</p> <p>Endpoints and RCiEPs are permitted to Set this bit to indicate that they implement the PTM Requester role.</p> <p>Switch Upstream Ports must Set this bit if the Switch contains one or more of the following:</p> <ul style="list-style-type: none"> <li>• A Downstream Port that implements the PTM Responder role.</li> <li>• An additional Function that implements the PTM Requester role.</li> </ul> <p>If a Device contains multiple Upstream Port Functions, the value of this bit must be consistent across all such Functions.</p>	<a href="#">HwInit</a>						
1	<p><b>PTM Responder Capable</b> - Root Ports and RCRBs are permitted to, and Switches supporting PTM must, Set this bit to indicate they implement the PTM Responder role (see <a href="#">Section 6.22.3.2</a>).</p> <p>If PTM Root Capable is Set, then this bit must be Set.</p>	<a href="#">HwInit</a>						
2	<p><b>PTM Root Capable</b> - Root Ports, RCRBs, and Switches are permitted to Set this bit if they are capable of being a source of PTM Master Time (see <a href="#">Section 6.22.1</a>).</p> <p>All other Functions must hardwire this bit to 0b.</p>	<a href="#">HwInit</a>						
3	<p><b>ePTM Capable</b> - If Set, indicates that this device supports Enhanced Precision Time Management (ePTM).</p> <p>It is strongly recommended that this bit be Set in all PTM Devices.</p>	<a href="#">HwInit</a>						
15:8	<p><b>Local Clock Granularity</b> - Encodings are:</p> <table> <tr> <td><b>0000 0000b</b></td> <td>Time Source does not implement a local clock. It simply propagates timing information obtained from further Upstream in the PTM Hierarchy when responding to PTM Request messages.</td> </tr> <tr> <td><b>0000 0001b to 1111 1110b</b></td> <td>Indicates the period of this Time Source's local clock in ns.</td> </tr> <tr> <td><b>1111 1111b</b></td> <td>Indicates the period of this Time Source's local clock is greater than 254 ns.</td> </tr> </table> <p>If the <u>PTM Root Select</u> bit is Set, this local clock is used to provide PTM Master Time. Otherwise, the Time Source uses this local clock to locally track PTM Master Time received from further Upstream within a PTM Hierarchy.</p> <p>This field is <u>RsvdP</u> if the <u>PTM Root Capable</u> bit is 0b.</p>	<b>0000 0000b</b>	Time Source does not implement a local clock. It simply propagates timing information obtained from further Upstream in the PTM Hierarchy when responding to PTM Request messages.	<b>0000 0001b to 1111 1110b</b>	Indicates the period of this Time Source's local clock in ns.	<b>1111 1111b</b>	Indicates the period of this Time Source's local clock is greater than 254 ns.	<a href="#">HwInit/RsvdP</a>
<b>0000 0000b</b>	Time Source does not implement a local clock. It simply propagates timing information obtained from further Upstream in the PTM Hierarchy when responding to PTM Request messages.							
<b>0000 0001b to 1111 1110b</b>	Indicates the period of this Time Source's local clock in ns.							
<b>1111 1111b</b>	Indicates the period of this Time Source's local clock is greater than 254 ns.							

### 7.9.16.3 PTM Control Register (Offset 08h)

This register controls a Function's participation in the Precision Time Measurement mechanism. Not all fields within this register apply to all Functions capable of implementing PTM.

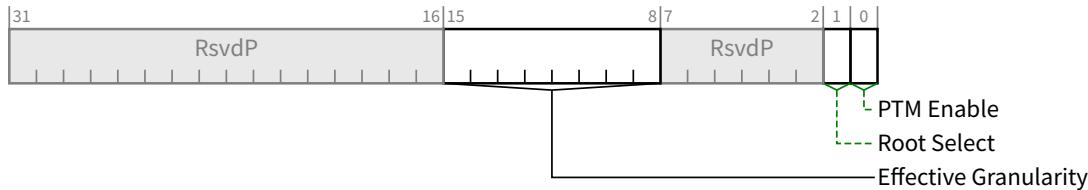


Figure 7-266 PTM Control Register

Table 7-216 PTM Control Register

Bit Location	Register Description	Attributes						
0	<p><b>PTM Enable</b> - When Set, this Function is permitted to participate in the PTM mechanism according to its selected role(s) (see Section 6.22.2 ).</p> <p>Default value is 0b.</p>	<u>RW</u>						
1	<p><b>Root Select</b> - When Set, if the <u>PTM Enable</u> bit is also Set, this Time Source is the PTM Root.</p> <p>Within each PTM Hierarchy, it is recommended that system software select only the furthest Upstream Time Source to be the PTM Root.</p> <p>Default value is 0b. If the value of the <u>PTM Root Capable</u> bit is 0b, this bit is permitted to be hardwired to 0b.</p>	<u>RW/RO</u>						
15:8	<p><b>Effective Granularity</b> - For Functions implementing the PTM Requester Role, this field provides information relating to the expected accuracy of the PTM clock, but does not otherwise affect the PTM mechanism.</p> <p>For Endpoints, system software must program this field to the value representing the maximum <u>Local Clock Granularity</u> reported by the PTM Root and all intervening PTM Time Sources.</p> <p>For RCiEPs, system software must set this field to the value reported in the <u>Local Clock Granularity</u> field by the associated PTM Time Source.</p> <p>Permitted values:</p> <table> <tr> <td><b>0000 0000b</b></td> <td>Unknown PTM granularity - one or more Switches between this Function and the PTM Root reported a <u>Local Clock Granularity</u> value of 0000 0000b.</td> </tr> <tr> <td><b>0000 0001b to 1111 1110b</b></td> <td>Indicates the effective PTM granularity in ns.</td> </tr> <tr> <td><b>1111 1111b</b></td> <td>Indicates the effective PTM granularity is greater than 254 ns.</td> </tr> </table> <p>Default value is 00000b. If <u>PTM Requester Capable</u> is Clear, this field is permitted to be hardwired to 0000 0000b.</p>	<b>0000 0000b</b>	Unknown PTM granularity - one or more Switches between this Function and the PTM Root reported a <u>Local Clock Granularity</u> value of 0000 0000b.	<b>0000 0001b to 1111 1110b</b>	Indicates the effective PTM granularity in ns.	<b>1111 1111b</b>	Indicates the effective PTM granularity is greater than 254 ns.	<u>RW/RO</u>
<b>0000 0000b</b>	Unknown PTM granularity - one or more Switches between this Function and the PTM Root reported a <u>Local Clock Granularity</u> value of 0000 0000b.							
<b>0000 0001b to 1111 1110b</b>	Indicates the effective PTM granularity in ns.							
<b>1111 1111b</b>	Indicates the effective PTM granularity is greater than 254 ns.							

## 7.9.17 Readiness Time Reporting Extended Capability

The Readiness Time Reporting Extended Capability provides an optional mechanism for describing the time required for a Device or Function to become Configuration-Ready. In the indicated situations, software is permitted to issue Requests to the Device or Function after waiting for the time advertised in this capability and need not wait for the (longer) times required elsewhere.

Software is permitted to issue requests upon the earliest of:

- Receiving a Readiness Notifications message (see Section 6.23 ).
- Waiting the appropriate time as specified in this document or in applicable specifications including the [PCI] and the [PCI-PM].
- Waiting the time indicated in the associated field of this capability.
- Waiting the time defined by system software or firmware<sup>157</sup> .

Software is permitted to cache values from this capability and to use those cached values as long as the same device operating in the same manner has not changed.

This capability is permitted to be implemented in all Functions.

A Function must be Configuration-Ready if:

- The Immediate Readiness bit is Clear and at least Reset Time has elapsed after the completion of Conventional Reset
  - If the Immediate Readiness bit is Set, Reset Time does not apply, and is Reserved
- The Function is associated with an Upstream Port and at least DL\_Up Time has elapsed after the Downstream Port above that Function reported Data Link Layer Link Active (see Section 7.5.3.8 ).
- The Function supports Function Level Reset and at least FLR Time has elapsed after that Function was issued a Function Level Reset.
- Immediate\_Readiness\_on\_Return\_to\_D0 is Clear and at least D3Hot to D0 Time has elapsed after that Function was directed to the D0 state from D3Hot.
  - If the Immediate\_Readiness\_on\_Return\_to\_D0 bit is Set, D3Hot to D0 Time does not apply, and is Reserved

When Immediate\_Readiness\_on\_Return\_to\_D0 is Clear, a Function must be Configuration-Ready when at least D3Hot to D0 Time has elapsed after the Function was directed to the D0 state from D3Hot. In addition, the Function must be in either the D0uninitialized or D0active state, depending on the value of the No\_Soft\_Reset bit.

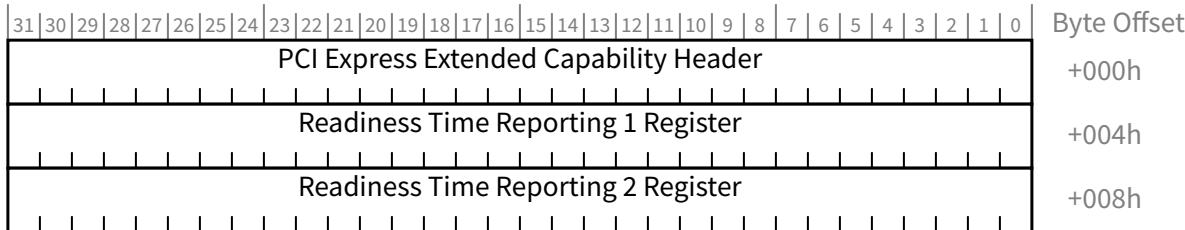
For VFs additional behavior is defined in Chapter 9 .

If the above conditions do not apply, Function behavior is not determined by the Readiness Time Reporting Extended Capability, and the Function must respond as defined elsewhere (including, for example, no response or a response with Configuration Retry Status).

The time values reported are determined by implementation-specific mechanisms. A Valid bit is defined in this capability to permit a device to defer reporting time values, for example to allow hardware initialization through driver-based mechanisms. If the Valid bit remains Clear and 1 minute has elapsed after device driver(s) have started, software is permitted to assume that no values will be reported.

<sup>157</sup>. For example, using ACPI tables to provide the equivalent of this capability.

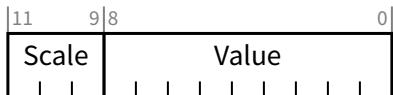
Registers and fields in the Readiness Time Reporting Extended Capability are shown in Figure 7-267. Time values are encoded in floating point as shown in Figure 7-268. The actual time value is  $\text{Value} \times \text{Multiplier}[\text{Scale}]$ . For example, the value A1Eh represents about 1 second (actually 1.006 sec) and the value 80Ah represents about 10 ms (actually 10.240 ms).



*Figure 7-267 Readiness Time Reporting Extended Capability*

Scale	Multiplier
0	1 ns
1	32 ns
2	1,024 ns
3	32,768 ns
4	1,048,576 ns
5	33,554,432 ns
6	Reserved
7	Reserved

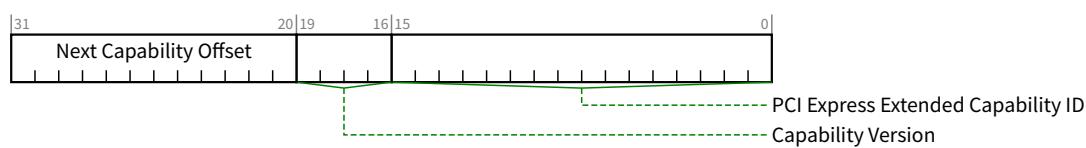
$\text{Multiplier} = 32^{\text{Scale}}$



*Figure 7-268 Readiness Time Encoding*

### 7.9.17.1 Readiness Time Reporting Extended Capability Header (Offset 00h)

Figure 7-269 and Table 7-217 detail allocation of fields in the Extended Capability header.



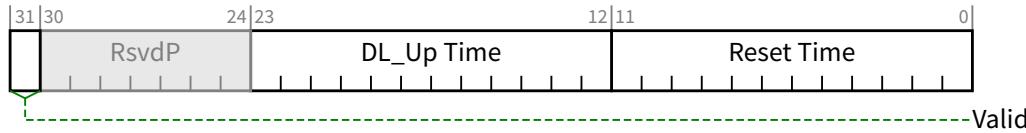
*Figure 7-269 Readiness Time Reporting Extended Capability Header*

*Table 7-217 Readiness Time Reporting Extended Capability Header*

Bit Location	Register Description	Attributes
15:0	<b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability.  Extended Capability ID for the <u>Readiness Time Reporting Extended Capability</u> is 0022h.	<u>RO</u>
19:16	<b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the Capability structure present.  Must be 1h for this version of the specification.	<u>RO</u>
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Capability structure or 000h if no other items exist in the linked list of Capabilities.  For Extended Capabilities implemented in Configuration Space, this offset is relative to the beginning of PCI-compatible Configuration Space and thus must always be either 000h (for terminating list of Capabilities) or greater than OFFh.	<u>RO</u>

### 7.9.17.2 Readiness Time Reporting 1 Register (Offset 04h)

Figure 7-270 and Table 7-218 detail allocation of fields in the Readiness Time Reporting 1 Register.

*Figure 7-270 Readiness Time Reporting 1 Register**Table 7-218 Readiness Time Reporting 1 Register*

Bit Location	Register Description	Attributes
11:0	<b>Reset Time</b> - is the time the Function requires to become <u>Configuration-Ready</u> after the completion of Conventional Reset.  This field is <u>RsvdP</u> if the <u>Immediate Readiness</u> bit is Set.  This field is undefined when the <u>Valid</u> bit is Clear.  This field must be less than or equal to the encoded value A1Eh.	<u>HwInit/RsvdP</u>
23:12	<b>DL Up Time</b> - is the time the Function requires to become <u>Configuration-Ready</u> after the Downstream Port above the Function reports <u>Data Link Layer Link Active</u> .  This field is <u>RsvdP</u> in Functions that are not associated with an Upstream Port.  This field is undefined when the <u>Valid</u> bit is Clear.  This field must be less than or equal to the encoded value A1Eh.	<u>HwInit/RsvdP</u>
31	<b>Valid</b> - If Set, indicates that all time values in this capability are valid. If Clear, indicates that the time values in this capability are not yet available.	<u>HwInit</u>

Bit Location	Register Description	Attributes
	<p>Time values may depend on device configuration. Device specific mechanisms, possibly involving the device driver(s), could be involved in determining time values.</p> <p>If this bit remains Clear and 1 minute has elapsed after all associated device driver(s) have started, software is permitted to assume that this bit will never be set.</p>	

### 7.9.17.3 Readiness Time Reporting 2 Register (Offset 08h)

Figure 7-271 and Table 7-219 detail allocation of fields in the Readiness Time Reporting 2 Register.

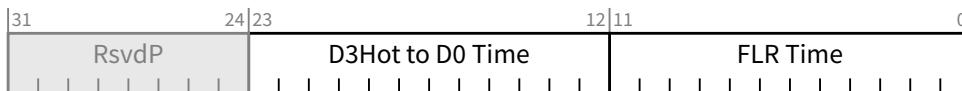


Figure 7-271 Readiness Time Reporting 2 Register

Table 7-219 Readiness Time Reporting 2 Register

Bit Location	Register Description	Attributes
11:0	<p><b>FLR Time</b> - is the time that the Function requires to become Configuration-Ready after it was issued an FLR.</p> <p>This field is RsvdP when the Function Level Reset Capability bit is Clear (see Section 7.5.3.3).</p> <p>This field is undefined when the Valid bit is Clear.</p> <p>This field must be less than or equal to the encoded value A1Eh.</p>	HwInit/RsvdP
23:12	<p><b>D3Hot to D0 Time</b> - If Immediate_Readyiness_on_Return_to_D0 is Clear, D3Hot to D0 Time is the time that the Function requires after it is directed from D3Hot to D0 before it is Configuration-Ready and has returned to either D0uninitialized or D0active state (see the PCI Bus Power Management Interface Specification).</p> <p>This field is RsvdP if the Immediate_Readyiness_on_Return_to_D0 bit is Set.</p> <p>This field is undefined when the Valid bit is Clear.</p> <p>This field must be less than or equal to the encoded value 80Ah.</p>	HwInit/RsvdP

### 7.9.18 Hierarchy ID Extended Capability

The Hierarchy ID Extended Capability provides an optional mechanism for passing a unique identifier to Functions within a Hierarchy. At most one instance of this capability is permitted in a Function. This capability is not applicable to Bridges, Root Complex Event Collectors, and RCRBs.

This capability takes three forms:

In Upstream Ports:

- This capability is permitted any Function associated with an Upstream Port.

- This capability is optional in Switch Upstream Ports. Support in Switch Upstream and Downstream Ports is independently optional.
- This capability is mandatory in Functions that use the Hierarchy ID Message. This includes use by the Function's driver.
- Functions, other than VFs, that have Hierarchy ID Writeable Clear, must report the Message Requester ID, Hierarchy ID, System GUID Authority ID, and System GUID fields from the most recently received Hierarchy ID Message.
- All VFs that have Hierarchy ID Writeable Clear, must report the same Hierarchy ID Valid, Message Requester ID, Hierarchy ID, System GUID Authority ID, and System GUID values as their associated PF.
- PFs must implement this capability if any of their VFs implement this capability.
- Functions that have Hierarchy ID Writeable Set must report the Hierarchy ID Valid, Message Requester ID, Hierarchy ID, System GUID Authority ID, and System GUID values programmed by software.

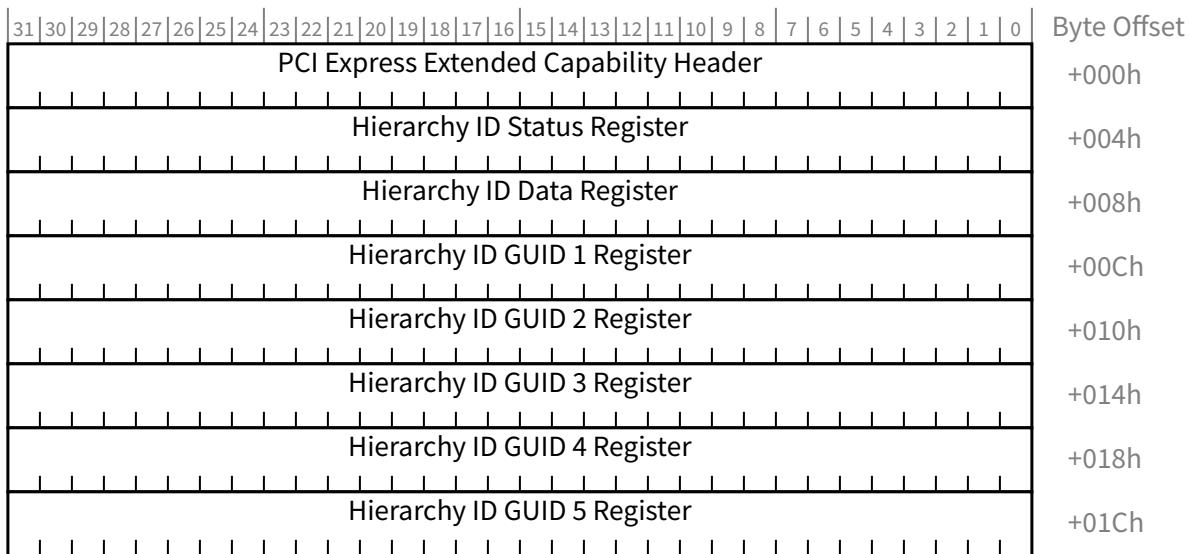
In Downstream Ports:

- This capability is permitted in any Downstream Port. It is recommended that it be implemented in Root Ports.
- When present in a Switch Downstream Port, this capability must be implemented in all Downstream Ports of the Switch. Support in Switch Upstream and Downstream Ports is independently optional.
- In Downstream Ports, the Hierarchy ID, System GUID Authority ID, and System GUID fields are Read / Write and contain the values to send in the Hierarchy ID Message.
- A Hierarchy ID capability is not affected by Hierarchy ID Messages forwarded through the associated Downstream Port.

In RCIEPs:

- VFs that have Hierarchy ID Writeable Clear must report the same Message Requester ID, Hierarchy ID, System GUID Authority ID, and System GUID values as their associated PF.
- PFs must implement this capability if any of their VFs implement this capability.
- Functions, other than VFs, that have Hierarchy ID Writeable Clear, must report the same Hierarchy ID Valid, Message Requester ID, Hierarchy ID, System GUID Authority ID, and System GUID values. The source of this information is outside the scope of this specification.
- Functions that have Hierarchy ID Writeable Set must report the Hierarchy ID Valid, Message Requester ID, Hierarchy ID, System GUID Authority ID, and System GUID values programmed by software.

Figure 7-272 details the layout of the Hierarchy ID Extended Capability.

*Figure 7-272 Hierarchy ID Extended Capability*

### 7.9.18.1 Hierarchy ID Extended Capability Header (Offset 00h)

Figure 7-273 and Table 7-220 detail allocation of fields in the Hierarchy ID Extended Capability Header.

*Figure 7-273 Hierarchy ID Extended Capability Header**Table 7-220 Hierarchy ID Extended Capability Header*

Bit Location	Description	Attributes
15:0	<b>Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability. PCI Express Extended Capability ID for the Hierarchy ID Extended Capability is 0028h.	RO
19:16	<b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the Capability structure present. Must be 1h for this version of the specification.	RO
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Extended Capability structure or 000h if no other items exist in the linked list of Capabilities. For Extended Capabilities in configuration space, this offset is relative to the beginning of PCI-compatible Configuration Space and thus must always be either 000h (for terminating the list of Capabilities) or greater than OFFh.	RO

### 7.9.18.2 Hierarchy ID Status Register (Offset 04h)

Figure 7-274 and Table 7-221 detail allocation of fields in the Hierarchy ID Status Register.

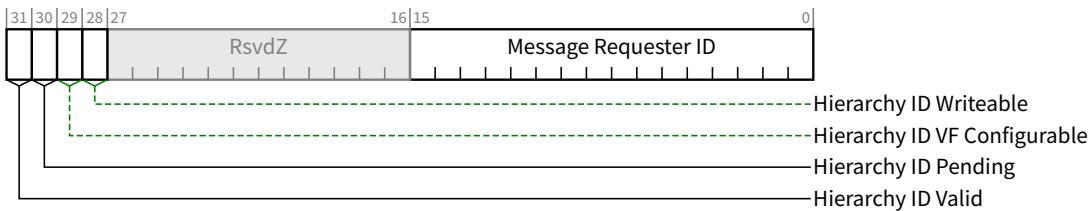


Figure 7-274 Hierarchy ID Status Register

Table 7-221 Hierarchy ID Status Register

Bit Location	Description	Attributes
15:0	<p><b>Message Requester ID</b> - In an Upstream Port, this field contains the Requester ID from the most recently received <u>Hierarchy ID Message</u>. This field is meaningful only if <u>Hierarchy ID Valid</u> is 1b. This value identifies the Downstream Port (within this Hierarchy) that sent the Hierarchy ID Message. This information is not considered part of the Hierarchy ID as it can vary within the Hierarchy (e.g., different Root Ports of one Root Complex), but helps in debug situations to identify the provenance of the Hierarchy ID information.</p> <p>In a Downstream Port, this field is <u>RsvdZ</u>.</p> <p>For <u>RCiEPs</u>, this field is <u>RsvdZ</u>.</p> <p>This field defaults to 0000h.</p>	RO/ <u>RsvdZ</u>
28	<p><b>Hierarchy ID Writeable</b> - This bit is Set to indicate that the <u>Hierarchy ID Data</u> and <u>GUID</u> registers are read/write. This bit is Clear to indicate that the <u>Hierarchy ID</u> and <u>GUID</u> registers are read only.</p> <p>In Downstream Ports this bit is hardwired to 1b.</p> <p>In Upstream Ports, Functions that are not VFs must hardwire this bit to 0b.</p> <p><u>RCiEPs</u> that are not VFs, must hardwire this bit to either 0b or 1b.</p> <p>VFs in an Upstream Port and Root Complex Integrated VFs are permitted to either:</p> <ul style="list-style-type: none"> <li>• hardwire this bit to 0b or</li> <li>• implement this bit as read / write with a default value of 0b.</li> </ul>	RW/RO
29	<p><b>Hierarchy ID VF Configurable</b> - This bit indicates that <u>Hierarchy ID Writeable</u> can be configured.</p> <p>If <u>Hierarchy ID Writeable</u> is implemented as read / write, this bit is 1b. Otherwise this bit is 0b.</p>	RO
30	<p><b>Hierarchy ID Pending</b> - In Downstream Ports this requests the transmission of a <u>Hierarchy ID Message</u>. Setting it requests transmission of a message based on the <u>Hierarchy Data</u> and <u>GUID</u> registers in this capability. This bit is cleared when either the transmit request is satisfied or the Link enters <u>DL_Down</u>. Behavior is undefined if the <u>Hierarchy Data</u> or <u>GUID</u> registers in this capability are written while this bit is Set.</p> <p>In Downstream Ports, this bit is Read / Write defaulting to 0b.</p> <p>In all other Functions, this bit is <u>RsvdZ</u>.</p>	RW/ <u>RsvdZ</u>

Bit Location	Description	Attributes
31	<p><b>Hierarchy ID Valid</b> - This bit indicates that the remaining fields in this capability are meaningful.</p> <p>In Downstream Ports, this bit is hardwired to 1b.</p> <p>In all other Functions, the following rules apply:</p> <ul style="list-style-type: none"> <li>If <u>Hierarchy ID Writeable</u> is Set, this bit is read/write, default 0b.</li> <li>If <u>Hierarchy ID Writeable</u> is Clear, this bit is read only, default 0b. <ul style="list-style-type: none"> <li>In VFs, this bit contains the same value as the associated PF.</li> <li>In Functions other than VFs that are associated with an Upstream Port, this bit is Set when a <u>Hierarchy ID Message</u> is received, and Cleared when the Link is DL_Down.</li> <li>In RCiEPs other than VFs, this bit contains a system provided value. The mechanism for determining this value is outside the scope of this specification.</li> </ul> </li> </ul>	RW/RO

### 7.9.18.3 Hierarchy ID Data Register (Offset 08h)

Figure 7-275 and Table 7-222 detail allocation of fields in the Hierarchy ID Data Register.

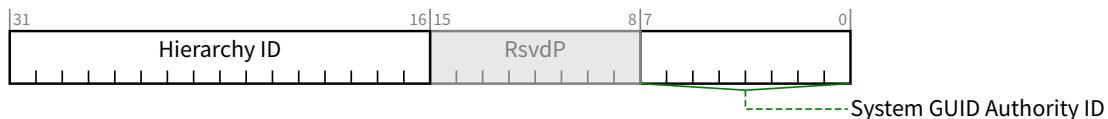


Figure 7-275 Hierarchy ID Data Register

Table 7-222 Hierarchy ID Data Register

Bit Location	Description	Attributes
7:0	<p><b>System GUID Authority ID</b> - This field corresponds to the <u>System GUID Authority ID</u> field in the <u>Hierarchy ID Message</u>. See Section 6.26 for details.</p> <p>This field is meaningful only if <u>Hierarchy ID Valid</u> is 1b.</p> <p>If <u>Hierarchy ID Writeable</u> is Set, this field is read-write and contains the value programmed by software.</p> <p>If <u>Hierarchy ID Writeable</u> is Clear, this field is read only. The value is determined using the rules defined in Section 7.9.18.</p> <p>This field defaults to 00h.</p>	RO/RW
31:16	<p><b>Hierarchy ID</b> - This field corresponds to the <u>Hierarchy ID</u> field in the <u>Hierarchy ID Message</u>. See Section 6.26 for details.</p> <p>This field is meaningful only if <u>Hierarchy ID Valid</u> is 1b.</p> <p>If <u>Hierarchy ID Writeable</u> is Set, this field is read-write and contains the value programmed by software.</p> <p>If <u>Hierarchy ID Writeable</u> is Clear, this field is read only. The value is determined using the rules defined in Section 7.9.18.</p> <p>This field defaults to 0000h.</p>	RO/RW

### 7.9.18.4 Hierarchy ID GUID 1 Register (Offset 0Ch)

Figure 7-276 and Table 7-223 detail allocation of fields in the Hierarchy ID GUID 1 Register.



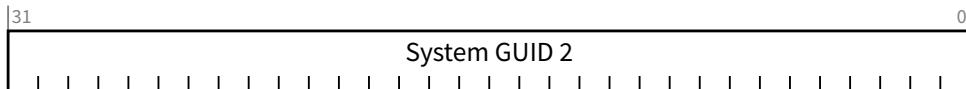
*Figure 7-276 Hierarchy ID GUID 1 Register*

*Table 7-223 Hierarchy ID GUID 1 Register*

Bit Location	Description	Attributes
15:0	<p><b>System GUID 1</b> - This field corresponds to bits [143:128] of the System GUID in the <u>Hierarchy ID Message</u>. See <u>Section 6.26</u> for details.</p> <p>This field is meaningful only if <u>Hierarchy ID Valid</u> is 1b.</p> <p>If <u>Hierarchy ID Writeable</u> is Set, this field is read-write and contains the value programmed by software.</p> <p>If <u>Hierarchy ID Writeable</u> is Clear, this field is read only. The value is determined using the rules defined in <u>Section 7.9.18</u>.</p> <p>This field defaults to 0000h.</p>	<u>RO/RW</u>

### 7.9.18.5 Hierarchy ID GUID 2 Register (Offset 10h)

Figure 7-277 and Table 7-224 detail allocation of fields in the Hierarchy ID GUID 2 Register.



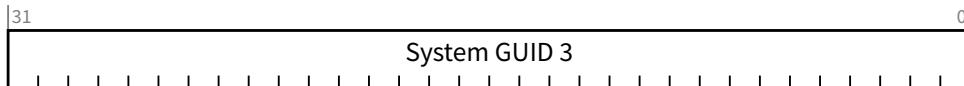
*Figure 7-277 Hierarchy ID GUID 2 Register*

*Table 7-224 Hierarchy ID GUID 2 Register*

Bit Location	Description	Attributes
31:0	<p><b>System GUID 2</b> - This field corresponds to bits [127:96] of the System GUID field in the <u>Hierarchy ID Message</u>. See <u>Section 6.26</u> for details.</p> <p>This field is meaningful only if <u>Hierarchy ID Valid</u> is 1b.</p> <p>If <u>Hierarchy ID Writeable</u> is Set, this field is read-write and contains the value programmed by software.</p> <p>If <u>Hierarchy ID Writeable</u> is Clear, this field is read only. The value is determined using the rules defined in <u>Section 7.9.18</u>.</p> <p>This field defaults to 0000 0000h.</p>	<u>RO/RW</u>

### 7.9.18.6 Hierarchy ID GUID 3 Register (Offset 14h)

Figure 7-278 and Table 7-225 detail allocation of fields in the Hierarchy ID GUID 3 Register.



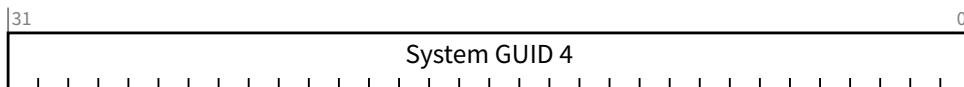
*Figure 7-278 Hierarchy ID GUID 3 Register*

*Table 7-225 Hierarchy ID GUID 3 Register*

Bit Location	Description	Attributes
31:0	<p><b>System GUID 3</b> - This field corresponds to bits [95:64] of the System GUID field in the <u>Hierarchy ID Message</u>. See <u>Section 6.26</u> for details.</p> <p>This field is meaningful only if <u>Hierarchy ID Valid</u> is 1b.</p> <p>If <u>Hierarchy ID Writeable</u> is Set, this field is read-write and contains the value programmed by software.</p> <p>If <u>Hierarchy ID Writeable</u> is Clear, this field is read only. The value is determined using the rules defined in <u>Section 7.9.18</u>.</p> <p>This field defaults to 0000 0000h.</p>	<u>RO/RW</u>

### 7.9.18.7 Hierarchy ID GUID 4 Register (Offset 18h)

Figure 7-279 and Table 7-226 detail allocation of fields in the Hierarchy ID GUID 4 Register.



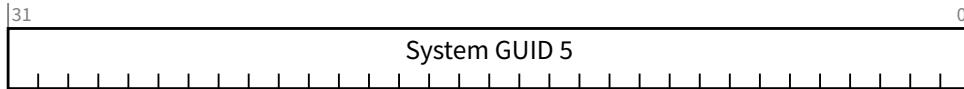
*Figure 7-279 Hierarchy ID GUID 4 Register*

*Table 7-226 Hierarchy ID GUID 4 Register*

Bit Location	Description	Attributes
31:0	<p><b>System GUID 4</b> - This field corresponds to bits [63:32] of the System GUID field in the <u>Hierarchy ID Message</u>. See <u>Section 6.26</u> for details.</p> <p>This field is meaningful only if <u>Hierarchy ID Valid</u> is 1b.</p> <p>If <u>Hierarchy ID Writeable</u> is Set, this field is read-write and contains the value programmed by software.</p> <p>If <u>Hierarchy ID Writeable</u> is Clear, this field is read only. The value is determined using the rules defined in <u>Section 7.9.18</u>.</p> <p>This field defaults to 0000 0000h.</p>	<u>RO/RW</u>

### 7.9.18.8 Hierarchy ID GUID 5 Register (Offset 1Ch)

Figure 7-280 and Table 7-227 detail allocation of fields in the Hierarchy ID GUID 5 Register.



*Figure 7-280 Hierarchy ID GUID 5 Register*

*Table 7-227 Hierarchy ID GUID 5 Register*

Bit Location	Description	Attributes
31:0	<p><b>System GUID 5</b> - This field corresponds to bits [31:0] of the System GUID field in the <u>Hierarchy ID Message</u>. See <u>Section 6.26</u> for details.</p> <p>This field is meaningful only if <u>Hierarchy ID Valid</u> is 1b.</p> <p>If <u>Hierarchy ID Writeable</u> is Set, this field is read-write and contains the value programmed by software.</p> <p>If <u>Hierarchy ID Writeable</u> is Clear, this field is read only. The value is determined using the rules defined in <u>Section 7.9.18</u>.</p> <p>This field defaults to 0000 0000h.</p>	<u>RO/RW</u>

### 7.9.19 Vital Product Data Capability (VPD Capability)

Support of VPD is optional. All Functions are permitted to contain the capability. This includes all Functions of a Multi-Function Device associated with an Upstream Port as well as RCiEPs.

Vital Product Data (VPD) is information that uniquely identifies hardware and, potentially, software elements of a system. The VPD can provide the system with information on various Field Replaceable Units such as part number, serial number, and other detailed information. The objective from a system point of view is to make this information available to the system owner and service personnel. VPD typically resides in a storage device (for example, a serial EEPROM) associated with the Function.

Details of the VPD Data is defined in Section 6.28.

Access to the VPD is provided using the Capabilities List in Configuration Space. The VPD Capability structure is shown in Figure 7-281.

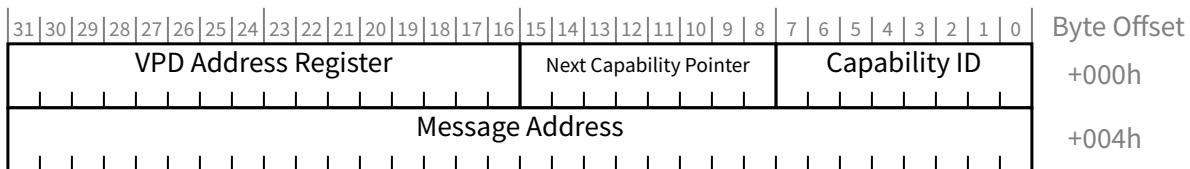


Figure 7-281 VPD Capability Structure

The following protocols are used to transfer data between the VPD Data field and the VPD storage component.

- To read VPD information:
  1. Issue single write to the VPD Address Register writing the flag bit (F) to 0b and VPD Address with the address to read.
  2. The hardware device will set F to 1b when 4 bytes of data from the storage component have been transferred to VPD Data.
  3. Software can monitor F and, after it becomes 1b, read the VPD information from VPD Data.

Behavior is undefined if either the VPD Address or VPD Data is written, prior to the flag bit becoming 1b.

- To write VPD information to the read/write portion of the VPD space:
  1. Write the data to VPD Data
  2. Then issue a single write to the VPD Address Register with F set to 1b and VPD Address set to the address where the VPD Data is to be stored.
  3. The software then monitors F and when it is set to 0b (by device hardware), the VPD Data (all 4 bytes) has been transferred from VPD Data to the storage component.

If either the VPD Address or VPD Data is written, prior to F being becoming 0b, the results of the write operation to the storage component are unpredictable.

Behavior is undefined if a read or write of the storage component is requested and VPD Address is outside the side of the storage component.

The VPD (both the read only items and the read/write fields) is stored information and will have no direct control of any device operations.

### 7.9.19.1 VPD Address Register

The VPD Address Register is used to request a read or write of the VPD storage component.

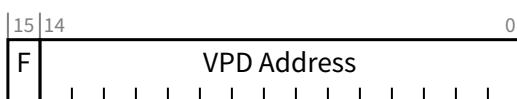
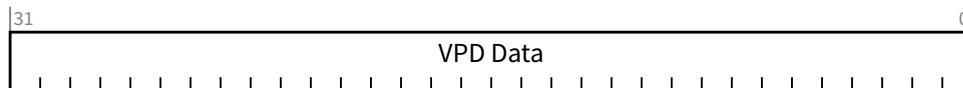


Figure 7-282 VPD Address Register

*Table 7-228 VPD Address Register*

Bit Location	Description	Attributes
14:0	<b>VPD Address</b> - DWORD-aligned byte address of the VPD to be accessed. Behavior is undefined if the lowest 2 bits of this field are non-zero. The lowest two bits of the field must be either RW, or RO with a value of 00b. The remaining bits of the field must be RW.  Default is implementation specific.	RW/RO (see description)
15	<b>F</b> - The F bit is always written along with <b>VPD Address</b> . The value of F indicates the direction of transfer being requested (0b = read, 1b = write). When the transfer is complete, the F bit value changes to indicate completion (1b = read complete, 0b = write complete).  Default is implementation specific.	RW

### 7.9.19.2 VPD Data Register

*Figure 7-283 VPD Data Register**Table 7-229 VPD Data Register*

Bit Location	Description	Attributes
31:0	<b>VPD Data</b> - <b>VPD Data</b> can be read through this register. The least significant byte of this register (at offset 04h in this capability structure) corresponds to the byte of VPD at the address specified by <b>VPD Address</b> . Behavior is undefined for any read or write of this register with Byte Enables other than 1111b.  Default is implementation specific.	RW

### 7.9.20 Native PCIe Enclosure Management Extended Capability (NPEM Extended Capability)

The Native PCIe Enclosure Management Extended (NPEM) Capability is an optional extended capability that is permitted to be implemented by Root Ports, Switch Downstream Ports, and Endpoints.

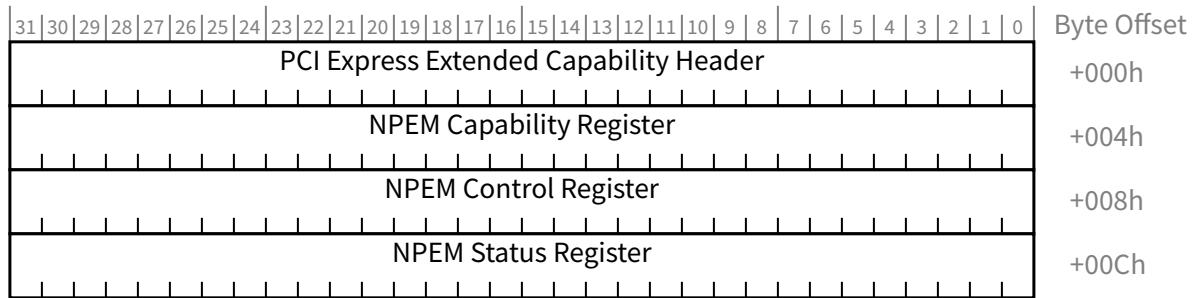


Figure 7-284 NPEM Extended Capability

### 7.9.20.1 NPEM Extended Capability Header (Offset 00h)



Figure 7-285 NPEM Extended Capability Header

Table 7-230 NPEM Extended Capability Header

Bit Location	Register Description	Attributes
15:0	<b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the extended capability. PCI Express Extended Capability ID for the NPEM Extended Capability is 0029h.	RO
19:16	<b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the capability structure present. Must be 1h for this version of the specification.	RO
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Extended Capability structure or 000h if no other items exist in the linked list of capabilities.	RO

### 7.9.20.2 NPEM Capability Register (Offset 04h)

The NPEM Capability Register contains an overall NPEM Capable bit and a bit map of states supported in the implementation. Implementations are required to support OK, Locate, Fail, and Rebuild states if NPEM Capable bit is Set. All other states are optional.

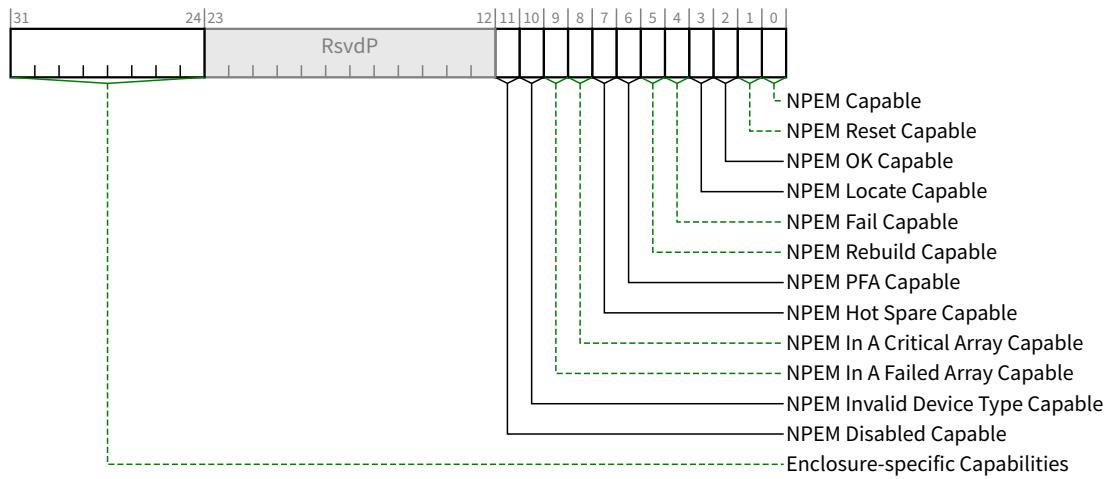


Figure 7-286 NPEM Capability Register

Table 7-231 NPEM Capability Register

Bit Location	Register Description	Attributes
0	<b>NPEM Capable</b> - When Set, this bit indicates that the enclosure has NPEM functionality.	<u>HwInit</u>
1	<b>NPEM Reset Capable</b> - A value of 1b indicates support for the optional NPEM Reset mechanism described in <a href="#">Section 6.29</a> . This capability is independently optional.	<u>HwInit</u>
2	<b>NPEM OK Capable</b> - When Set, this bit indicates that enclosure has the ability to indicate the NPEM OK state. This bit must be Set if <u>NPEM Capable</u> is also Set.	<u>HwInit</u>
3	<b>NPEM Locate Capable</b> - When Set, this bit indicates that enclosure has the ability to indicate the NPEM Locate state. This bit must be Set if <u>NPEM Capable</u> is also Set.	<u>HwInit</u>
4	<b>NPEM Fail Capable</b> - When Set, this bit indicates that enclosure has the ability to indicate the NPEM Fail state. This bit must be Set if <u>NPEM Capable</u> is also Set.	<u>HwInit</u>
5	<b>NPEM Rebuild Capable</b> - When Set, this bit indicates that enclosure has the ability to indicate the NPEM Rebuild state. This bit must be Set if <u>NPEM Capable</u> is also Set.	<u>HwInit</u>
6	<b>NPEM PFA Capable</b> - When Set, this bit indicates that enclosure has the ability to indicate the NPEM PFA state. This capability is independently optional.	<u>HwInit</u>
7	<b>NPEM Hot Spare Capable</b> - When Set, this bit indicates that enclosure has the ability to indicate the NPEM Hot Spare state. This capability is independently optional.	<u>HwInit</u>
8	<b>NPEM In A Critical Array Capable</b> - When Set, this bit indicates that enclosure has the ability to indicate the NPEM In A Critical Array state. This capability is independently optional.	<u>HwInit</u>
9	<b>NPEM In A Failed Array Capable</b> - When Set, this bit indicates that enclosure has the ability to indicate the NPEM In A Failed Array state. This capability is independently optional.	<u>HwInit</u>
10	<b>NPEM Invalid Device Type Capable</b> - When Set, this bit indicates that enclosure has the ability to indicate the NPEM_Invalid_Device_Type state. This capability is independently optional.	<u>HwInit</u>

Bit Location	Register Description	Attributes
11	<b>NPEM Disabled Capable</b> - When Set, this bit indicates that enclosure has the ability to indicate the NPEM_Disabled state. This capability is independently optional.	<u>HwInit</u>
31:24	<b>Enclosure-specific Capabilities</b> - The definition of enclosure-specific bits is outside the scope of this specification.	<u>HwInit</u>

### 7.9.20.3 NPEM Control Register (Offset 08h)

The NPEM Control Register contains an overall NPEM Enable bit and a bit map of states that software controls.

Use of Enclosure-specific bits is outside the scope of this specification.

All writes to this register, including writes that do not change the register value, are NPEM commands and should eventually result in a command completion indication in the NPEM Status Register.

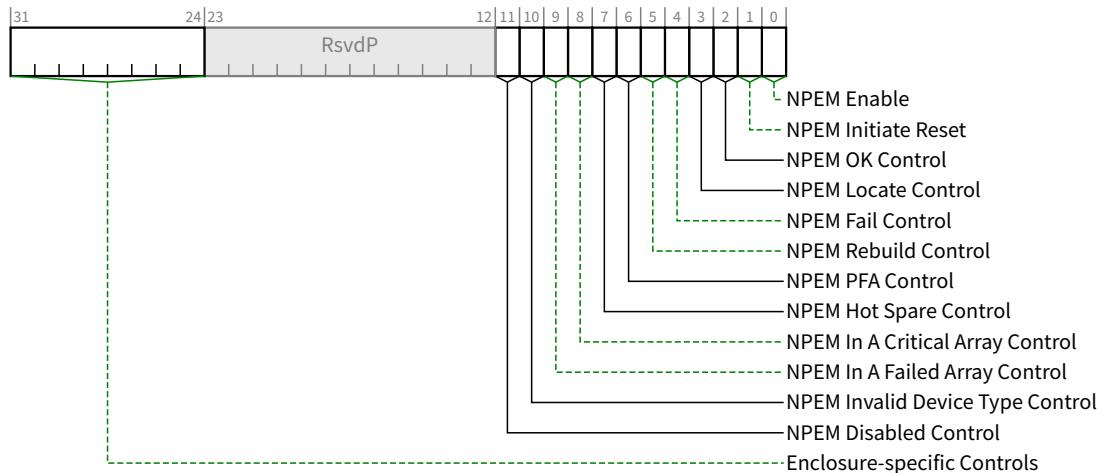


Figure 7-287 NPEM Control Register

Table 7-232 NPEM Control Register

Bit Location	Register Description	Attributes
0	<b>NPEM Enable</b> - When Set, this bit enables the NPEM capability. When Clear, this bit disables the NPEM capability. Default value of this bit is 0b. When enabled, this capability operates as defined in this specification. When disabled, the other bits in this capability have no effect and any associated indications are outside the scope of this specification.	<u>RW</u>
1	<b>NPEM Initiate Reset</b> - If NPEM Reset Capable bit is 1b, then a write of 1b to this bit initiates NPEM Reset. If NPEM Reset Capable bit is 0b, then this bit is permitted to be read-only with a value of 0b. The value read by software from this bit must always be 0b.	<u>RW/RO</u>
2	<b>NPEM OK Control</b> - When Set, this bit specifies that the NPEM OK indication be turned ON. When Clear, this bit specifies that the NPEM OK indication be turned OFF.	<u>RW/RO</u>

Bit Location	Register Description	Attributes
	<p>If NPEM OK Capable bit in <u>NPEM Capability Register</u> is 0b, this bit is permitted to be read-only with a value of 0b.</p> <p>Default value of this bit is 0b</p>	
3	<p><b>NPEM Locate Control</b> - When Set, this bit specifies that the NPEM Locate indication be turned ON. When Clear, this bit specifies that the NPEM Locate indication be turned OFF.</p> <p>If NPEM Locate Capable bit in the <u>NPEM Capability Register</u> is 0b, this bit is permitted to be read-only with a value of 0b.</p> <p>Default value of this bit is 0b</p>	<u>RW/RO</u>
4	<p><b>NPEM Fail Control</b> - When Set, this bit specifies that the NPEM Fail indication be turned ON. When Clear, this bit specifies that the NPEM Fail indication be turned OFF.</p> <p>If NPEM Fail Capable bit in the <u>NPEM Capability Register</u> is 0b, this bit is permitted to be read-only with a value of 0b.</p> <p>Default value of this bit is 0b</p>	<u>RW/RO</u>
5	<p><b>NPEM Rebuild Control</b> - When Set, this bit specifies that the NPEM Rebuild indication be turned ON. When Clear, this bit specifies that the NPEM Rebuild indication be turned OFF.</p> <p>If NPEM Rebuild Capable bit in <u>NPEM Capability Register</u> is 0b, this bit is permitted to be read-only with a value of 0b.</p> <p>Default value of this bit is 0b</p>	<u>RW/RO</u>
6	<p><b>NPEM PFA Control</b> - When Set, this bit specifies that the NPEM PFA indication be turned ON. When Clear, this bit specifies that the NPEM PFA indication be turned OFF.</p> <p>If NPEM PFA Capable bit in <u>NPEM Capability Register</u> is 0b, this bit is permitted to be read-only with a value of 0b.</p> <p>Default value of this bit is 0b</p>	<u>RW/RO</u>
7	<p><b>NPEM Hot Spare Control</b> - When Set, this bit specifies that the NPEM Hot Spare indication be turned ON. When Clear, this bit specifies that the NPEM Hot Spare indication be turned OFF.</p> <p>If NPEM Hot Spare Capable bit in <u>NPEM Capability Register</u> is 0b, this bit is permitted to be read-only with a value of 0b.</p> <p>Default value of this bit is 0b</p>	<u>RW/RO</u>
8	<p><b>NPEM In A Critical Array Control</b> - When Set, this bit specifies that the NPEM In A Critical Array indication be turned ON. When Clear, this bit specifies that the NPEM In A Critical Array indication be turned OFF.</p> <p>If NPEM In A Critical Array Capable bit in <u>NPEM Capability Register</u> is 0b, this bit is permitted to be read-only with a value of 0b.</p> <p>Default value of this bit is 0b</p>	<u>RW/RO</u>
9	<p><b>NPEM In A Failed Array Control</b> - When Set, this bit specifies that the NPEM In A Failed Array indication be turned ON. When Clear, this bit specifies that the NPEM In A Failed Array indication be turned OFF.</p> <p>If NPEM In A Failed Array Capable bit in <u>NPEM Capability Register</u> is 0b, this bit is permitted to be read-only with a value of 0b.</p> <p>Default value of this bit is 0b</p>	<u>RW/RO</u>
10	<p><b>NPEM Invalid Device Type Control</b> - When Set, this bit specifies that the NPEM Invalid Device Type indication be turned ON. When Clear, this bit specifies that the NPEM Invalid Device Type indication be turned OFF.</p>	<u>RW/RO</u>

Bit Location	Register Description	Attributes
	If <u>NPEM Invalid Device Type Capable</u> bit in <u>NPEM Capability Register</u> is 0b, this bit is permitted to be read-only with a value of 0b.  Default value of this bit is 0b	
11	<b>NPEM Disabled Control</b> - When Set, this bit specifies that the NPEM Disabled indication be turned ON. When Clear, this bit specifies that the NPEM Disabled indication be turned OFF.  If <u>NPEM Disabled Capable</u> bit in <u>NPEM Capability Register</u> is 0b, this bit is permitted to be read-only with a value of 0b.  Default value of this bit is 0b	<u>RW/RO</u>
31:24	<b>Enclosure-specific Controls</b> - The definition of enclosure-specific bits is outside the scope of this specification. Enclosure-specific software is permitted to change the value of this field. Other software must preserve the existing value when writing this register.  Default value of this field is 00h	<u>RW/RO</u>

#### 7.9.20.4 NPEM Status Register (Offset 0Ch)

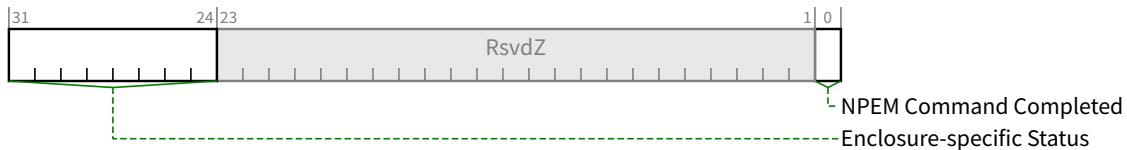


Figure 7-288 NPEM Status Register

Table 7-233 NPEM Status Register

Bit Location	Register Description	Attributes
0	<b>NPEM Command Completed</b> - This bit is Set when an NPEM command has completed, and the NPEM controller is ready to accept a subsequent command.  This bit is permitted to be hardwired to 1b if the enclosure is able to accept writes that update any portion of the NPEM Control register without any delay between successive writes.  Default value of this bit is 0b.  Software must wait for an NPEM command to complete before issuing the next NPEM command. However, if this bit is not set within 1 second limit on command execution, software is permitted to repeat the NPEM command or issue the next NPEM command. If software issues a write before the Port has completed processing of the previous command and before the 1 second time limit has expired, the Port is permitted to either accept or discard the write. Such a write is considered a programming error, and could result in a discrepancy between the <u>NPEM Control Register</u> and the enclosure element state. To recover from such a programming error and return the enclosure to a consistent state, software must issue a write to the <u>NPEM Control Register</u> which conforms to the NPEM command completion rules.	<u>RW1C / RO</u>
31:24	<b>Enclosure-specific Status</b> - The definition of enclosure specific bits is outside the scope of this specification. Enclosure specific software is permitted to write non-zero values to this field. Other software must write 00h to this field.	<u>RsvdZ/RO/RW1C</u>

Bit Location	Register Description	Attributes
	The default value of this field is enclosure-specific. This field is permitted to be hardwired to 00h.	

## 7.9.21 Alternate Protocol Extended Capability

The Alternate Protocol Extended Capability structure is optional. It is only permitted in:

- A Function associated with a Downstream Port.
- Function 0 (and only Function 0) of a Device associated with an Upstream Port.

Figure 7-289 details allocation of register fields in the Alternate Protocol Extended Capability structure.

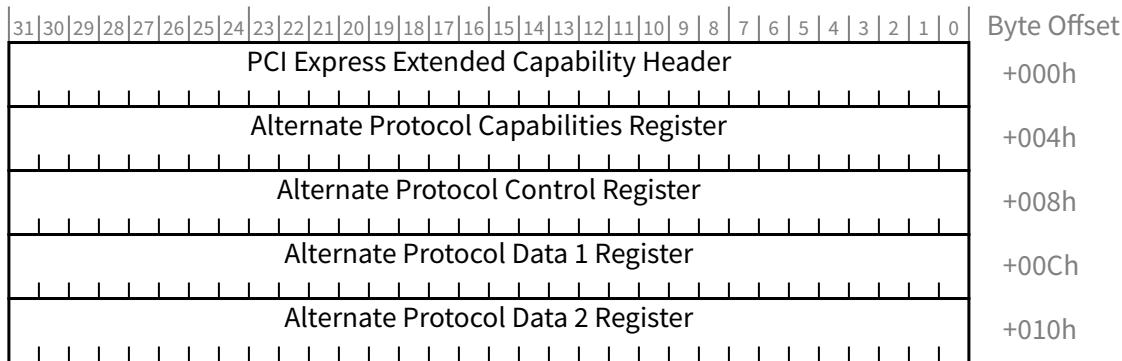


Figure 7-289 Alternate Protocol Extended Capability

### 7.9.21.1 Alternate Protocol Extended Capability Header (Offset 00h)

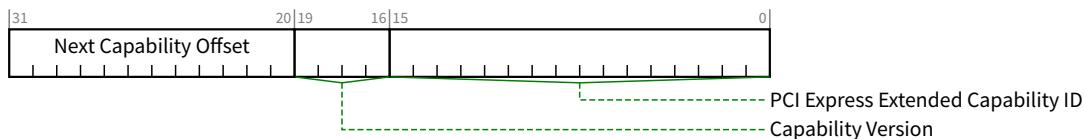


Figure 7-290 Alternate Protocol Extended Capability Header

Table 7-234 Alternate Protocol Extended Capability Header

Bit Location	Register Description	Attributes
15:0	<b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability.  The Extended Capability ID for the Alternate Protocol Capability is 002Bh.	RO

Bit Location	Register Description	Attributes
19:16	<p><b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the Capability structure present.</p> <p>Must be 1h for this version of the specification.</p>	RO
31:20	<p><b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Capability structure or 000h if no other items exist in the linked list of Capabilities.</p> <p>For Extended Capabilities implemented in Configuration Space, this offset is relative to the beginning of PCI-compatible Configuration Space and thus must always be either 000h (for terminating list of Capabilities) or greater than OFFh.</p>	RO

### 7.9.21.2 Alternate Protocol Capabilities Register (Offset 04h)

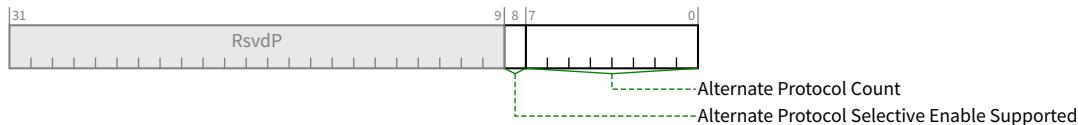


Figure 7-291 Alternate Protocol Capabilities Register

Table 7-235 Alternate Protocol Capabilities Register

Bit Location	Register Description	Attributes
7:0	<p><b>Alternate Protocol Count</b> - Indicates the number of Alternate Protocols supported by one or more Lanes of this Link.</p> <p>Since support for PCI Express is mandatory, the value of this field must be greater than or equal to 1.</p>	HwInit
8	<p><b>Alternate Protocol Selective Enable Supported</b> - If Set, the <u>Alternate Protocol Selective Enable Mask Register</u> is present. If Clear, the <u>Alternate Protocol Selective Enable Mask Register</u> is not present and Alternate Protocol Negotiation is controlled solely by the <u>Alternate Protocol Negotiation Global Enable bit</u>.</p> <p>In Upstream Ports, this bit is hardwired to 0b.</p> <p>In Downstream Ports, this bit is HwInit with an implementation specific default value.</p>	RO/HwInit

### 7.9.21.3 Alternate Protocol Control Register (Offset 08h)

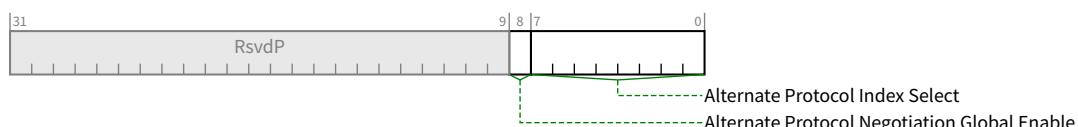


Figure 7-292 Alternate Protocol Control Register

Table 7-236 Alternate Protocol Control Register

Bit Location	Register Description	Attributes
7:0	<p><b>Alternate Protocol Index Select</b> - This field determines which Lane and which Alternate Protocol of that Lane is visible in <u>Alternate Protocol Data 1 Register</u> and <u>Alternate Protocol Data 2 Register</u>.</p> <p>The default value of this field is 00h. Unused bits in this field are permitted to be hardwired to 0b.</p> <p>If <u>Alternate Protocol Count</u> is 01h, this field is permitted to be hardwired to 00h.</p> <p>Behavior is undefined if this field is greater than <u>Alternate Protocol Count</u>.</p> <p>Specific <u>Alternate Protocol Index Select</u> values are permitted to be disabled without renumbering other protocol index values. Disabled entries return an <u>Alternate Protocol Vendor ID</u> of FFFFh.</p>	RW
8	<p><b>Alternate Protocol Negotiation Global Enable</b> - When this bit is Set, Alternate Protocol Negotiation is enabled for this Link. When this bit is Clear, Alternate Protocol Negotiation is disabled for this Link.</p> <p>Default is 0b.</p>	RW

#### 7.9.21.4 Alternate Protocol Data 1 Register (Offset 0Ch)

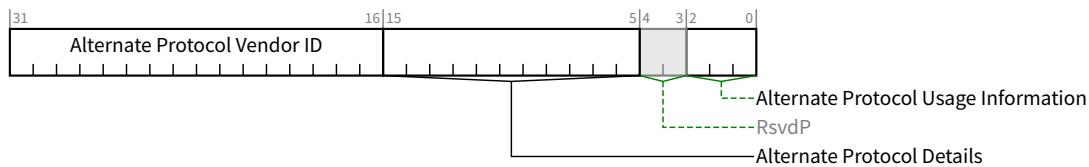


Figure 7-293 Alternate Protocol Data 1 Register

Table 7-237 Alternate Protocol Data 1 Register

Bit Location	Register Description	Attributes
2:0	<p><b>Alternate Protocol Usage Information</b> - This field contains the <u>Modified TS Usage</u> associated alternate protocol associated with the <u>Alternate Protocol Index Select</u> value.</p> <p>If <u>Alternate Protocol Vendor ID</u> is FFFFh, the value of this field is undefined.</p>	RO
15:5	<p><b>Alternate Protocol Details</b> - This field contains the <u>Alternate Protocol Details</u> associated alternate protocol associated with the <u>Alternate Protocol Index Select</u> value.</p> <p>If <u>Alternate Protocol Vendor ID</u> is FFFFh, the value of this field is undefined.</p>	RO
31:16	<p><b>Alternate Protocol Vendor ID</b> - This field contains the Vendor ID associated alternate protocol associated with the <u>Alternate Protocol Index Select</u> value.</p> <p>Bits 7:0 of this field contain bits 7:0 of Vendor ID (Symbol 10).</p> <p>Bits 15:8 of this field contain bits 15:8 of Vendor ID (Symbol 11).</p> <p>If <u>Alternate Protocol Index Select</u> is greater than or equal to <u>Alternate Protocol Count</u>, this field contains FFFFh.</p> <p>If <u>Alternate Protocol Index Select</u> is associated with a disabled alternate protocol, this field contains FFFFh.</p>	RO

### 7.9.21.5 Alternate Protocol Data 2 Register (Offset 10h)

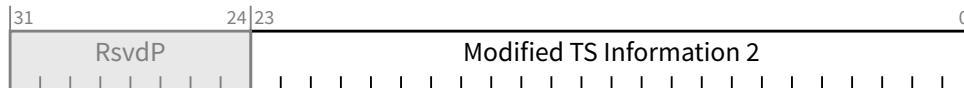


Figure 7-294 Alternate Protocol Data 2 Register

Table 7-238 Alternate Protocol Data 2 Register

Bit Location	Register Description	Attributes
23:0	<p><b>Modified TS Information 2</b> - This field contains the value for symbols 12 through 14 for the alternate protocol associated with the <u>Alternate Protocol Index Select</u> value.</p> <p>If <u>Alternate Protocol Vendor ID</u> is FFFFh, the value of this field is undefined.</p> <p>Bits 7:0 contain the value of Symbol 12.</p> <p>Bits 16:8 contain the value of Symbol 13.</p> <p>Bits 23:16 contain the value of Symbol 14.</p>	RO

### 7.9.21.6 Alternate Protocol Selective Enable Mask Register (Offset 14h)

This register is present if Alternate Protocol Selective Enable Supported is Set.

This register consists of a bit mask of size Alternate Protocol Count bits. Each bit corresponds to a valid value of Alternate Protocol Index Select. This register is an integral number of DWORDs in size.

When Alternate Protocol Negotiation Global Enable is Set, a particular bit in this register is Set, and the corresponding Alternate Protocol is not disabled (see Alternate Protocol Index Select), the next Alternate Protocol negotiation is permitted to consider using that Alternate Protocol. When a particular bit in this register is Clear, the next Alternate Protocol negotiation is not permitted to consider using the corresponding Alternate Protocol.

Changes to this field will affect the next Alternate Protocol negotiation and have no effect on current operation of the Link (regardless of current protocol).

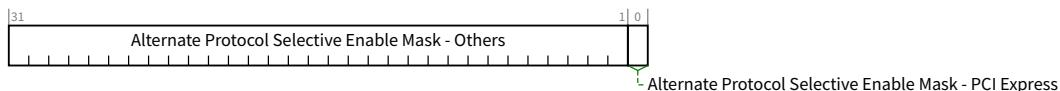


Figure 7-295 Alternate Protocol Selective Enable Mask Register

Table 7-239 Alternate Protocol Selective Enable Mask Register

Bit Location	Register Description	Attributes
0	<p><b>Alternate Protocol Selective Enable Mask - PCI Express</b> - The PCI Express Protocol is always index 00h. The default value of this bit is 1b (i.e., PCI Express is always enabled by default).</p>	RWS

Bit Location	Register Description	Attributes
	This bit is permitted to be hardwired to 1b.	
31:1	<p><b>Alternate Protocol Selective Enable Mask - Others</b> - Other bits in this register represent protocols other than PCI Express. The default values of these “other” bits is implementation specific.</p> <p>The width of this field is shown here as 32 bits. The actual width depends on <u>Alternate Protocol Count</u>.</p> <p>Bits in this field corresponding to disabled Alternate Protocol Index values are permitted to be hardwired to 0b.</p> <p>Bits in this field corresponding to <u>Alternate Protocol Index Select</u> values above <u>Alternate Protocol Count</u> are permitted to be hardwired to 0b.</p>	RWS

## 7.9.22 Conventional PCI Advanced Features Capability (AF)

This capability is optional. It is permitted only in Conventional PCI Functions that are integrated into a Root Complex. A Function may contain at most one instance of this capability.

Figure 7-296 shows the layout of this capability.

Note: Due to document production limitations, this figure shows an 8 byte capability while the actual capability is only 6 bytes long. Bytes 6 and 7 in the figure are not part of the capability.

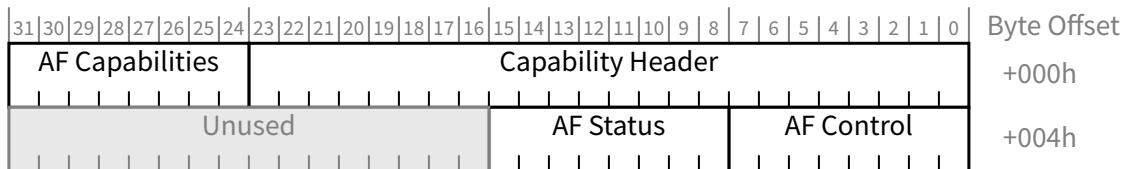


Figure 7-296 Conventional PCI Advanced Features Capability (AF)

### 7.9.22.1 Advanced Features Capability Header (Offset 00h)

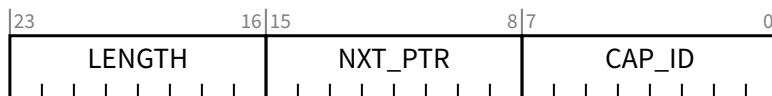


Figure 7-297 Advanced Features Capability Header

Table 7-240 Advanced Features Capability Header

Bit Location	Register Description	Attributes
7:0	<b>CAP_ID</b> - The value of 13h in this field identifies the Function as being AF capable.	RO

Bit Location	Register Description	Attributes
15:8	<b>NXT_PTR</b> - Pointer to the next item in the capabilities list. Must be 00h for the final item in the list.	<u>RO</u>
23:16	<b>LENGTH</b> - AF Structure Length (Bytes). Shall return a value of 06h.	<u>RO</u>

### 7.9.22.2 AF Capabilities Register (Offset 03h)

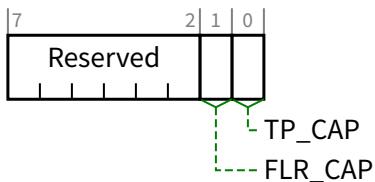


Figure 7-298 AF Capabilities Register

Table 7-241 AF Capabilities Register

Bit Location	Register Description	Attributes
0	<b>TP_CAP</b> - Set to indicate support for the <u>Transactions Pending</u> (TP) bit. TP_CAP must be Set if FLR_CAP is Set.	<u>HwInit</u>
1	<b>FLR_CAP</b> - Set to indicate support for <u>Function Level Reset</u> (INITIATE_FLR).	<u>HwInit</u>
7:2	Reserved - Shall be implemented as read only returning a value of 000 0000b.	<u>RO</u>

### 7.9.22.3 Conventional PCI Advanced Features Control Register (Offset 04h)



Figure 7-299 Conventional PCI Advanced Features Control Register

Table 7-242 Conventional PCI Advanced Features Control Register

Bit Location	Register Description	Attributes
0	<b>Function Level Reset (INITIATE_FLR)</b> - A write of 1b initiates a Function Level Reset (FLR). Registers and state information that do not apply to Conventional PCI are exempt from the FLR requirements in this specification (see <u>Section 6.6.2</u> ).  The value read by software from this bit shall always be 0b.	<u>RW</u>

Bit Location	Register Description	Attributes
7:1	Reserved - Shall be implemented as read only returning a value of 000 0000b.	<u>RO</u>

#### 7.9.22.4 AF Status Register (Offset 05h)

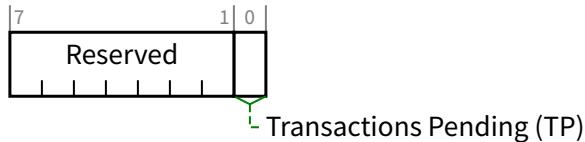


Figure 7-300 AF Status Register

Table 7-243 AF Status Register

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

#### 7.9.23 SFI Extended Capability

The SFI (System Firmware Intermediary) Extended Capability is an optional capability that provides system firmware with enhanced control over primarily hot-plug mechanisms, and enables system firmware to operate as an intermediary between certain events and the operating system (see [Section 6.7.4](#)). This capability may be implemented by a Root Port or a Switch Downstream Port. It is not applicable to any other Device/Port type.

If a Downstream Port implements the SFI Extended Capability, that Port must support ERR\_COR Subclass capability, and indicate so by Setting the [ERR\\_COR Subclass Capable](#) bit in the Device Capabilities Register. See [see Section 7.5.3.3](#).

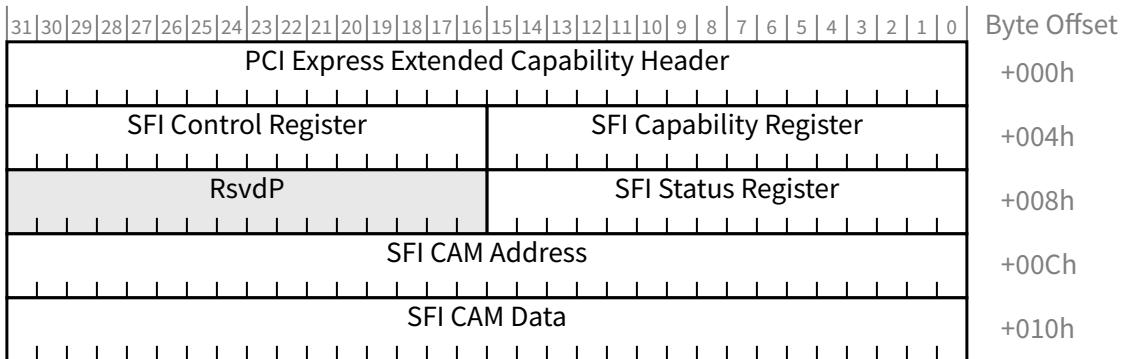


Figure 7-301 SFI Extended Capability

### 7.9.23.1 SFI Extended Capability Header (Offset 00h)

Figure 7-302 and Table 7-244 detail allocation of fields in the Extended Capability header.

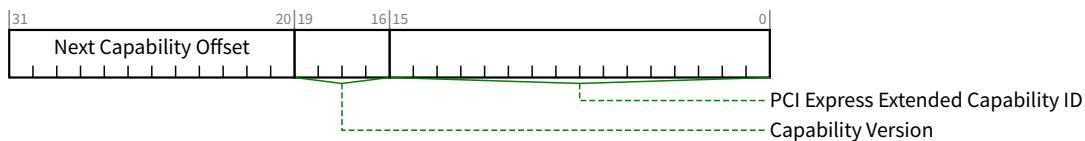


Figure 7-302 SFI Extended Capability Header

Table 7-244 SFI Extended Capability Header

Bit Location	Register Description	Attributes
15:0	<b>PCI Express Extended Capability ID</b> - This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability. Extended Capability ID for the SFI Extended Capability is 002Ch.	RO
19:16	<b>Capability Version</b> - This field is a PCI-SIG defined version number that indicates the version of the Capability structure present. Must be 1h for this version of the specification.	RO
31:20	<b>Next Capability Offset</b> - This field contains the offset to the next PCI Express Capability structure or 000h if no other items exist in the linked list of Capabilities. For Extended Capabilities implemented in Configuration Space, this offset is relative to the beginning of PCI-compatible Configuration Space and thus must always be either 000h (for terminating list of Capabilities) or greater than OFFh.	RO

### 7.9.23.2 SFI Capability Register (Offset 04h)

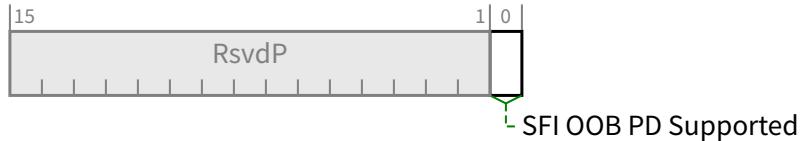


Figure 7-303 SFI Capability Register

Table 7-245 SFI Capability Register

Bit Location	Register Description	Attributes
0	<b>SFI OOB PD Supported</b> - When Set, this bit indicates that this slot supports reporting the out-of-band presence detect state. If this Downstream Port has no implemented slot (as indicated by the <u>Slot Implemented</u> bit in the PCI Express Capabilities Register), then the value of this bit must be 0b.	<u>HwInit</u>

### 7.9.23.3 SFI Control Register (Offset 06h)

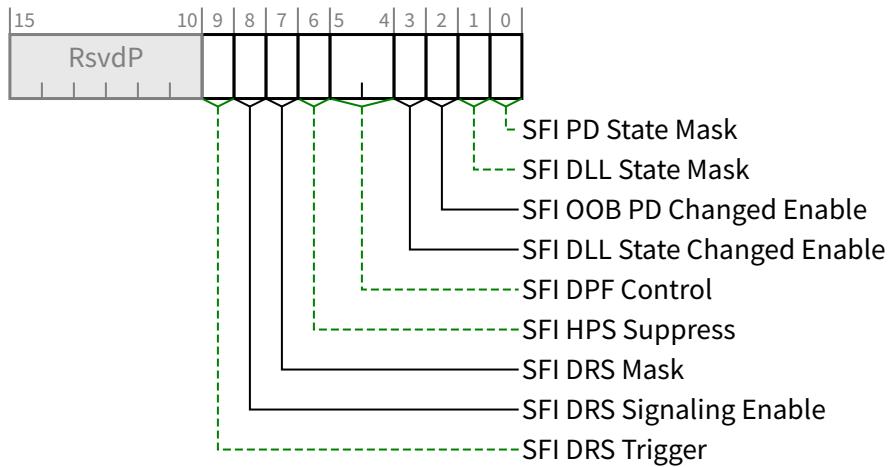


Figure 7-304 SFI Control Register

Table 7-246 SFI Control Register

Bit Location	Register Description	Attributes
0	<b>SFI PD State Mask</b> - When Set, this bit masks the <u>Presence Detect State</u> bit in the <u>Slot Status Register</u> , making its value 0b, regardless of the actual presence detect state. Otherwise, its value indicates the actual state.	<u>RW</u>

Bit Location	Register Description	Attributes
	If the value of the <u>Presence Detect State</u> bit changes when the <u>SFI PD State Mask</u> bit value changes, this must cause a <u>Presence Detect Changed</u> event (see <u>Section 6.7.3</u> ).  Default value of this bit is 0b.	
1	<b>SFI DLL State Mask</b> - When Set, this bit masks the <u>Data Link Layer Link Active</u> bit in the <u>Link Status</u> Register, making its value 0b, regardless of the actual Data Link Layer state. Otherwise, its value indicates the actual state.  If the value of the <u>Data Link Layer Link Active</u> bit changes when the <u>SFI DLL State Mask</u> bit value changes, this must cause a <u>Data Link Layer State Changed</u> event (see <u>Section 6.7.3</u> ).  Default value of this bit is 0b.	<u>RW</u>
2	<b>SFI OOB PD Changed Enable</b> - When Set, this bit enables sending an <u>ERR_COR</u> Message for the <u>SFI OOB PD Changed</u> event. See <u>Section 6.7.4.1</u> for other necessary conditions.  This bit must be <u>RW</u> if the <u>SFI OOB PD Supported</u> bit is Set; otherwise, it is permitted to be hardwired to 0b. If the <u>SFI OOB PD Supported</u> bit is Clear and software Sets this bit, the behavior is undefined.  Default value of this bit is 0b.	<u>RW/RO</u>
3	<b>SFI DLL State Changed Enable</b> - When Set, this bit enables sending an <u>ERR_COR</u> Message for the <u>SFI DLL State Changed</u> event. See <u>Section 6.7.4.1</u> for other necessary conditions.  Default value of this bit is 0b.	<u>RW</u>
5:4	<b>SFI DPF Control</b> - This field controls the level of Downstream Port Filtering (DPF) enabled on the Downstream Port, governing which Request TLPs targeting Downstream Components get filtered; that is, handled as if the Link is in <u>DL_Down</u> . See <u>Section 6.7.4.2</u> .  Defined encodings are:  <b>00b</b> Disabled <b>01b</b> Filter all Request TLPs <b>10b</b> Filter only Configuration Request TLPs <b>11b</b> Reserved  Default value of this field is 00b.	<u>RW</u>
6	<b>SFI HPS Suppress</b> - When Set, this bit forces the <u>Hot-Plug Surprise</u> (HPS) bit in the <u>Slot Capabilities</u> Register to be Clear and disables associated Hot-Plug Surprise functionality. See <u>Section 6.7.4.4</u> .  Default value of this bit is 0b.	<u>RW</u>
7	<b>SFI DRS Mask</b> - When Set, this bit masks the <u>DRS Message Received</u> bit in the <u>Link Status 2</u> Register, making its value 0b, regardless of the actual DRS Message Received state. Otherwise, its value indicates the actual state.  If the value of the <u>DRS Message Received</u> bit changes from Clear to Set when the <u>SFI DRS Mask</u> bit is Cleared, this must trigger any notification enabled by the <u>DRS Signaling Control</u> field in the <u>Link Control Register</u> (see <u>Section 7.5.3.7</u> ).  Default value of this bit is 0b.	<u>RW</u>
8	<b>SFI DRS Signaling Enable</b> - When Set, this bit enables sending an <u>ERR_COR</u> Message for the <u>SFI DRS Received</u> event. See <u>Section 6.7.4.1</u> for other necessary conditions.  Default value of this bit is 0b.	<u>RW</u>
9	<b>SFI DRS Trigger</b> - If the <u>SFI DRS Mask</u> bit is Clear, when software writes a 1b to this bit, the Downstream Port must behave as if a DRS Message was received. Otherwise, software writing a 1b to this bit has no effect.	<u>RW</u>

Bit Location	Register Description	Attributes
	<p>It is permitted to write 1b to this bit while simultaneously writing updated values to other fields in this register, notably the <a href="#">SFI DRS Mask</a> bit. For this case, the <a href="#">SFI DRS Trigger</a> semantics are based on the updated value of the <a href="#">SFI DRS Mask</a> bit.</p> <p>This bit always returns 0b when read.</p>	

#### 7.9.23.4 SFI Status Register (Offset 08h)

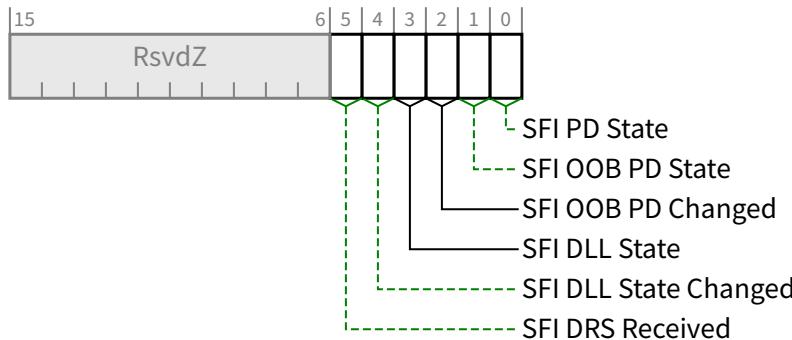


Figure 7-305 SFI Status Register

Table 7-247 SFI Status Register

Bit Location	Register Description	Attributes
0	<b>SFI PD State</b> - This bit always indicates the actual presence detect state associated with the <a href="#">Presence Detect State</a> bit in the <a href="#">Slot Status Register</a> , even when the value of that bit is being masked by the <a href="#">SFI PD State Mask</a> bit.	<a href="#">RO</a>
1	<b>SFI OOB PD State</b> - This bit indicates the out-of-band presence detect state, independent of the in-band presence detect state.  This bit must be implemented if the <a href="#">SFI OOB PD Supported</a> bit is Set; otherwise, it is permitted to be hardwired to 0b.	<a href="#">RO</a>
2	<b>SFI OOB PD Changed</b> - This bit is Set when the value reported in the <a href="#">SFI OOB PD State</a> bit is changed.	<a href="#">RW1C</a>
3	<b>SFI DLL State</b> - This bit always indicates the actual link state associated with the <a href="#">Data Link Layer Link Active</a> bit in the <a href="#">Link Status Register</a> , even when the value of that bit is being masked by the <a href="#">SFI DLL State Mask</a> bit.	<a href="#">RO</a>
4	<b>SFI DLL State Changed</b> - This bit is Set when the value reported in the <a href="#">SFI DLL State</a> bit is changed.	<a href="#">RW1C</a>
5	<b>SFI DRS Received</b> - This bit always indicates the actual state associated with the <a href="#">DRS Message Received</a> bit in the <a href="#">Link Status 2 Register</a> , even when the value of that bit is being masked by the <a href="#">SFI PD State Mask</a> bit.  Clearing the <a href="#">SFI DRS Received</a> bit (by writing a 1b to it) must also cause the actual state associated with the <a href="#">DRS Message Received</a> bit to be Cleared.	<a href="#">RW1C</a>

### 7.9.23.5 SFI CAM Address Register (Offset 0Ch)



Figure 7-306 SFI CAM Address Register

Table 7-248 SFI CAM Address Register

Bit Location	Register Description	Attributes
27:0	<b>SFI CAM Address</b> - This field specifies the target Bus, Device, and Function Numbers, along with the Extended Register Number and Register Number, in the format specified by Table 7-1 .	RW

### 7.9.23.6 SFI CAM Data Register (Offset 10h)



Figure 7-307 SFI CAM Data Register

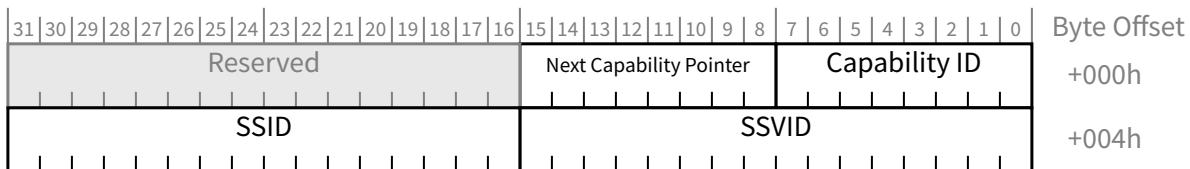
Table 7-249 SFI CAM Data Register

Bit Location	Register Description	Attributes
31:0	<b>SFI CAM Data</b> - When this field is read, the SFI CAM generates and transmits a Configuration Read Request on the Link below this Port. When this field is written, the SFI CAM generates and transmits a Configuration Write Request on the Link below this Port. In both cases, the target of the Configuration Request is determined by the value of the <u>SFI CAM Address Register</u> . See <u>Section 6.7.4.3</u> .	RW

### 7.9.24 Subsystem ID and Sybsystem Vendor ID Capability

The Subsystem ID and Sybsystem Vendor ID Capability is an optional capability used to uniquely identify the add-in card or subsystem where the PCI device resides. It provides a mechanism for add-in card vendors to distinguish their add-in cards from one another even though the add-in cards may have the same PCI bridge on them (and, therefore, the same Vendor ID and Device ID). The format of the capability is shown in Figure 7-308 . The fields are described in Table 7-250 .

This capability is only permitted in Type 1 Configuration Space Headers.

Figure 7-308 Subsystem ID and Sybsystem Vendor ID CapabilityTable 7-250 Subsystem ID and Sybsystem Vendor ID Capability

Bit Location	Register Description	Attributes
DWORD 0 Bits 7:0	<b>Capability ID</b> - Indicates the PCI Express Capability structure. This field must return a Capability ID of 0Dh indicating that this is a <u>Subsystem ID and Sybsystem Vendor ID Capability</u> structure.	<u>RO</u>
DWORD 0 Bits 15:8	<b>Next Capability Pointer</b> - This field contains the offset to the next PCI Capability structure or 00h if no other items exist in the linked list of Capabilities.	<u>RO</u>
DWORD 1 Bits 15:0	<b>SSVID</b> - The SSVID identifies the manufacturer of the add-in card or subsystem. The SSVID is assigned by PCI-SIG to insure uniqueness (the Vendor ID is used as the SSVID also). This field is read-only.	<u>HwInit</u>
DWORD 1 Bits 31:16	<b>SSID</b> - The SSID identifies the particular add-in card or subsystem and is assigned by the vendor. This field is read-only.	<u>HwInit</u>

