

Figure 7-13 Expansion ROM Base Address Register

Table 7-9 Expansion ROM Base Address Register

Bit Location	Description	Attributes
0	<p><b>Expansion ROM Enable</b> - This bit controls whether or not the Function accepts accesses to its Expansion ROM via the Expansion ROM Base Address Register. Functions that support an Expansion ROM accessible through this register must implement this bit. If the Function has an Enhanced Allocation Capability that includes an EA entry for an Expansion ROM, this bit must be hardwired to 0b (see Section 7.5.1.2.4). Functions that do not support an Expansion ROM are permitted to hardwire this bit to 0b. When this bit is 0b, the Function's Expansion ROM address space via the Expansion ROM Base Address Register is disabled. When the bit is 1b, address decoding is enabled using the Expansion ROM Base Address field in this register. This optionally allows a Function to be used with or without an Expansion ROM depending on system configuration. The Memory Space Enable bit in the Command register has precedence over the Expansion ROM Enable bit. A Function must claim accesses to its Expansion ROM via the Expansion ROM Base Address Register only if both the Memory Space Enable bit and the Expansion ROM Enable bit are Set. The default value of this bit is 0b.</p> <p>In order to minimize the number of address decoders needed, a Function is permitted to share a decoder between the Expansion ROM Base Address Register and other Base Address registers or entry entries in the Enhanced Allocation Capability<sup>142</sup>. When Expansion ROM Enable is Set, the decoder is used for accesses to the Expansion ROM and device independent software must not access the Function through any other Base Address registers or entry in the Enhanced Allocation Capability. Address decode sharing is not permitted for PFs (see Section 9.3.4.1.15) or if the Function contains an Enhanced Allocation Capability with an EA entry for an Expansion ROM.</p>	RO/RW
3:1	<p><b>Expansion ROM Validation Status</b> - Expansion ROM Validation is optional. When this field is non-zero, it indicates the status of hardware validation of the Expansion ROM contents.</p> <ul style="list-style-type: none"> <li>An Expansion ROM is considered valid if it passes an implementation-specific integrity check.</li> <li>An Expansion ROM is considered valid-warn if the implementation-specific integrity check passes but indicates an implementation-specific warning condition.</li> <li>A valid or valid-warn Expansion ROM is also considered trusted if passes an optional implementation-specific trust test (e.g., signed by a trusted certificate).</li> <li>Hardware validation must include the contents of the Expansion ROM. This validation status is also permitted to cover additional internal information (e.g., internal firmware). Validation does not include Vital Product Data (see Section 6.28).</li> <li>It is optional whether an implementation is capable of returning Validation Status values 011b, 101b, 110b, or 111b.</li> </ul> <p>Defined encodings are:</p>	HwInit/ROS

142. Note that it is the address decoder that is shared, not the registers themselves. The Expansion ROM Base Address Register and other Base Address registers or entries in the Enhanced Allocation Capability must be able to hold unique values at the same time.

Bit Location	Description	Attributes
	<p><b>000b</b> Validation not supported</p> <p><b>001b</b> Validation in Progress</p> <p><b>010b</b> Validation Pass Valid contents, trust test was not performed</p> <p><b>011b</b> Validation Pass Valid and trusted contents</p> <p><b>100b</b> Validation Fail Invalid contents</p> <p><b>101b</b> Validation Fail Valid but untrusted contents (e.g., Out of Date, Expired or Revoked Certificate)</p> <p><b>110b</b> Warning Pass Validation Passed with implementation-specific warning. Valid contents, trust test was not performed</p> <p><b>111b</b> Warning Pass Validation Passed with implementation-specific warning. Valid and trusted contents</p> <ul style="list-style-type: none"> <li>• If the Function does not support validation, this field must be hardwired to 000b.</li> <li>• If the Function supports validation and has an Enhanced Allocation Capability with an EA entry for an Expansion ROM, this field is HwInit and its value must be between 010b and 111b (see Section 7.8.5.3).</li> <li>• Otherwise, this field is Read Only Sticky and has a default value of 001b. When validation completes, this field must contain a value between 010b and 111b inclusive.</li> <li>• Software is permitted to assume validation will never complete if this field contains 001b and 1 minute has passed after de-assertion of Fundamental Reset. This field is only reset by Fundamental Reset, and is not affected by other resets.</li> </ul>	
7:4	<p><b>Expansion ROM Validation Details</b> - contains optional, implementation-specific details associated with Expansion ROM Validation.</p> <ul style="list-style-type: none"> <li>• If the Function does not support validation, this field is RsvdP.</li> <li>• This field is optional. When validation is supported and this field is not implemented, this field must be hardwired to 0000b. Any unused bits in this field are permitted to be hardwired to 0b.</li> <li>• If validation is in progress (Expansion ROM Validation Status is 001b), non-zero values of this field represent implementation-specific indications of the phase of the validation progress (e.g., 50% complete). The value 0000b indicates that no validation progress information is provided.</li> <li>• If validation is completed (Expansion ROM Validation Status 010b to 111b inclusive), non-zero values in this field represent additional implementation-specific information. The value 0000b indicates that no information is provided.</li> <li>• If the Function supports validation and has an Enhanced Allocation Capability with an EA entry for an Expansion ROM, this field is HwInit.</li> <li>• Otherwise, this field is Read Only Sticky. This field is only reset by Fundamental Reset, and is not affected by other resets.</li> <li>• This field must not change value once the validation process completes.</li> <li>• It is recommended that system software include the value of this field when it reports validation status (e.g., error log).</li> </ul>	HwInit/ROS/RsvdP
31:11	<p><b>Expansion ROM Base Address</b> - contains the upper bits 21 bits of the starting memory address of the Expansion ROM. The lower 11 bits of the Expansion ROM Base Address Register are masked off (set to zero) by software to form a 32-bit address.</p>	RW/RO

Bit Location	Description	Attributes
	<p>This field functions like the address portion of a 32-bit Base Address register. This field corresponds to the upper 21 bits of the <u>Expansion ROM Base Address</u>. The number of bits (out of these 21) that a Function actually implements depends on how much Expansion ROM address space the Function requires. For instance, a Function that requires a 64 KB area to map its Expansion ROM would implement the top 16 bits in this field as writeable, leaving the bottom 5 bits (out of these 21) hardwired to 0b. The amount of address space a Function requests must not be greater than 16 MB. Functions that support an Expansion ROM accessible through this register must implement this field. If the Function has an <u>Enhanced Allocation Capability</u> that includes an EA entry for an Expansion ROM, this field must be hardwired to 0 (see Section 7.8.5.3 ) Functions that do not support an Expansion ROM are permitted to hardwire this field to 0.</p> <p>Device independent configuration software can determine how much address space the Function requires by writing a value of all 1's to this field and then reading the value back. The Function will return 0's in all don't-care bits, effectively specifying the size and alignment requirements. The amount of address space a Function requests must not be greater than 16 MB.</p>	

### 7.5.1.2.5 ***Min\_Gnt Register/Max\_Lat Register (Offset 3Eh/3Fh)***

These registers do not apply to PCI Express. They must be read-only and hardwired to 00h.

### 7.5.1.3 ***Type 1 Configuration Space Header***

Figure 7-14 details allocation for register fields of Type 1 Configuration Space Header for Switch and Root Ports.

Byte Offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																	
	Device ID														Vendor ID																																		
+000h	Status														Command																																		
+004h	Class Code														Revision ID																																		
+008h	BIST	Header Type							Primary Latency Timer							Cache Line Size																																	
+00Ch	Base Address Register 0																																																
+010h	Base Address Register 1																																																
+014h	Secondary Latency Timer	Subordinate Bus Number	Secondary Bus Number	Primary Bus Number																																													
+018h	Secondary Status							I/O Limit							I/O Base																																		
+01Ch	Memory Limit														Memory Base																																		
+020h	Prefetchable Memory Limit														Prefetchable Memory Base																																		
+024h	Prefetchable Memory Base Upper 32 Bits																																																
+028h	Prefetchable Memory Limit Upper 32 Bits																																																
+02Ch	I/O Base Limit 16 Bits														U/O Base Upper 16 Bits																																		
+030h	Reserved														Capabilities Pointer																																		
+034h	Expansion ROM Base Address																																																
+038h	Bridge Control														Interrupt Pin							Interrupt Line																											
+03Ch																																																	

Figure 7-14 [Type 1 Configuration Space Header](#)

Section 7.5.1.1 details the PCI Express-specific registers that are valid for all Configuration Space Header types. The PCI Express-specific interpretation of registers specific to Type 1 Configuration Space Header is defined in this section.

Register interpretations described in this section apply to PCI-PCI Bridge structures representing Switch and Root Ports; other device Functions such as PCI Express to PCI/PCI-X Bridges with Type 1 Configuration Space headers are not covered by this section.

### 7.5.1.3.1 Type 1 Base Address Registers (Offset 10h-14h)

These registers are defined in Section 7.5.1.2.1. However the number of BARs available within the Type 1 Configuration Space Header is different than that of the Type 0 Configuration Space Header.

### **7.5.1.3.2 Primary Bus Number Register (Offset 18h)**

Except as noted, this register is not used by PCI Express Functions but must be implemented as read-write and the default value must be 00h, for compatibility with legacy software. PCI Express Functions capture the Bus (and Device) Number as described (including exceptions) in [Section 2.2.6](#). Refer to [\[PCIe-to-PCI-PCI-X-Bridge\]](#) for exceptions to this requirement.

### **7.5.1.3.3 Secondary Bus Number Register (Offset 19h)**

The Secondary Bus Number register is used to record the bus number of the PCI bus segment to which the secondary interface of the bridge is connected. Configuration software programs the value in this register. The Bridge uses this register to determine when and how to respond to an ID-routed TLP observed on its primary interface, notably when to forward the TLP to its secondary interface, in certain cases after performing some conversion. See [Section 7.3.3](#) for Configuration Request routing and conversion rules. This register must be implemented as read/write and the default value must be 00h.

### **7.5.1.3.4 Subordinate Bus Number Register (Offset 1Ah)**

The Subordinate Bus Number register is used to record the bus number of the highest numbered PCI bus segment which is behind (or subordinate to) the bridge. Configuration software programs the value in this register. The Bridge uses this register to determine when and how to respond to an ID-routed TLP observed on its primary interface, notably when to forward the TLP to its secondary interface. See [Section 7.3.3](#) for Configuration Request routing rules. This register must be implemented as read-write and the default value must be 00h.

### **7.5.1.3.5 Secondary Latency Timer (Offset 1Bh)**

This register does not apply to PCI Express. It must be read-only and hardwired to 00h. For PCI Express to PCI/PCI-X Bridges, refer to the [\[PCIe-to-PCI-PCI-X-Bridge\]](#) for requirements for this register.

### **7.5.1.3.6 I/O Base/I/O Limit Registers(Offset 1Ch/1Dh)**

The I/O Base and I/O Limit registers are optional and define an address range that is used by the bridge to determine when to forward I/O transactions from one interface to the other.

If a bridge does not implement an I/O address range, then both the I/O Base and I/O Limit registers must be implemented as read-only registers that return zero when read. If a bridge supports an I/O address range, then these registers must be initialized by configuration software so default states are not specified.

If a bridge implements an I/O address range, the upper 4 bits of both the I/O Base and I/O Limit registers are writable and correspond to address bits Address[15:12]. For the purpose of address decoding, the bridge assumes that the lower 12 address bits, Address[11:0], of the I/O base address (not implemented in the I/O Base register) are zero. Similarly, the bridge assumes that the lower 12 address bits, Address[11:0], of the I/O limit address (not implemented in the I/O Limit register) are FFFh. Thus, the bottom of the defined I/O address range will be aligned to a 4 KB boundary and the top of the defined I/O address range will be one less than a 4 KB boundary.

The I/O Limit register can be programmed to a smaller value than the I/O Base register, if there are no I/O addresses on the secondary side of the bridge. In this case, the bridge will not forward any I/O transactions from the primary bus to the secondary and will forward all I/O transactions from the secondary bus to the primary bus.

The lower four bits of both the I/O Base and I/O Limit registers are read-only, contain the same value, and encode the I/O addressing capability of the bridge according to Table 7-10.

*Table 7-10 I/O Addressing Capability*

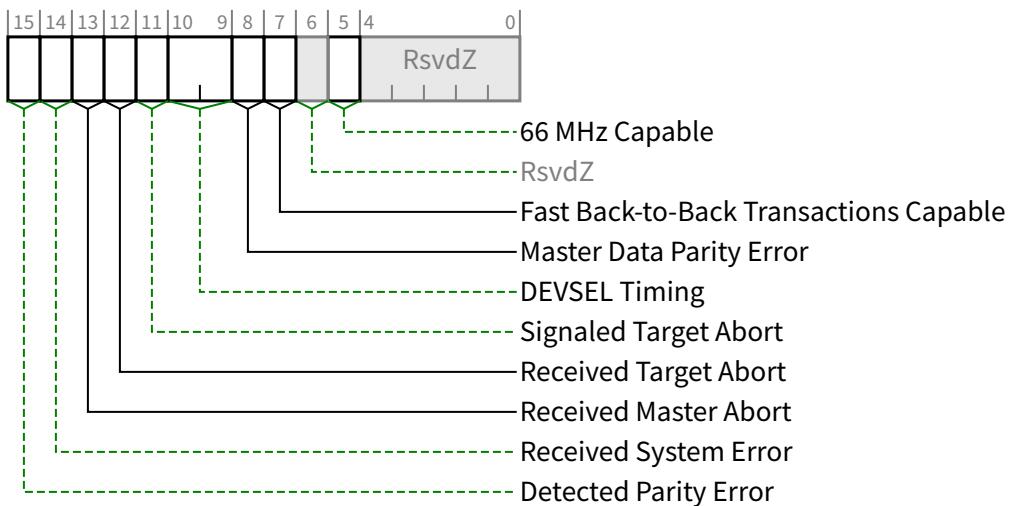
Bits 3:0	I/O Addressing Capability
0h	16-bit I/O addressing
1h	32-bit I/O addressing
2h-Fh	Reserved

If the low four bits of the I/O Base and I/O Limit registers have the value 0000b, then the bridge supports only 16-bit I/O addressing (for ISA compatibility), and, for the purpose of address decoding, the bridge assumes that the upper 16 address bits, Address[31:16], of the I/O base and I/O limit address (not implemented in the I/O Base and I/O Limit registers) are zero. Note that the bridge must still perform a full 32-bit decode of the I/O address (i.e., check that Address[31:16] are 0000h). In this case, the I/O address range supported by the bridge will be restricted to the first 64 KB of I/O Space (0000 0000h to 0000 FFFFh).

If the low four bits of the I/O Base and I/O Limit registers are 0001b, then the bridge supports 32-bit I/O address decoding, and the I/O Base Upper 16 Bits and the I/O Limit Upper 16 Bits hold the upper 16 bits, corresponding to Address[31:16], of the 32-bit I/O Base and I/O Limit addresses respectively. In this case, system configuration software is permitted to locate the I/O address range supported by the bridge anywhere in the 4 GB I/O Space. Note that the 4 KB alignment and granularity restrictions still apply when the bridge supports 32-bit I/O addressing.

### 7.5.1.3.7 Secondary Status Register (Offset 1Eh)

Table 7-11 defines the Secondary Status Register and Figure 7-15 provides the register layout. For PCI Express to PCI/PCI-X Bridges, refer to the [PCIe-to-PCI-PCI-X-Bridge] for requirements for this register.



*Figure 7-15 Secondary Status Register*

Table 7-11 Secondary Status Register

Bit Location	Register Description	Attributes
5	<b>66 MHz Capable</b> - This bit was originally described in the [PCI-to-PCI-Bridge]. Its functionality does not apply to PCI Express and the bit must be hardwired to 0b.	<u>RO</u>
7	<b>Fast Back-to-Back Transactions Capable</b> - This bit was originally described in the [PCI-to-PCI-Bridge]. Its functionality does not apply to PCI Express and the bit must be hardwired to 0b.	<u>RO</u>
8	<p><b>Master Data Parity Error</b> - See <a href="#">Section 7.5.1.1.14</a>.</p> <p>This bit is Set by a Function with a <a href="#">Type 1 Configuration Space Header</a> if the <a href="#">Parity Error Response Enable</a> bit in the <a href="#">Bridge Control Register</a> is Set and either of the following two conditions occurs:</p> <ul style="list-style-type: none"> <li>Port receives a Poisoned Completion coming Upstream</li> <li>Port transmits a Poisoned Request Downstream</li> </ul> <p>If the <a href="#">Parity Error Response Enable</a> bit is Clear, this bit is never Set.</p> <p>Default value of this bit is 0b.</p>	<u>RW1C</u>
10:9	<b>DEVSEL Timing</b> - This field was originally described in the [PCI-to-PCI-Bridge]. Its functionality does not apply to PCI Express and the field must be hardwired to 00b.	<u>RO</u>
11	<p><b>Signaled Target Abort</b> - See <a href="#">Section 7.5.1.1.14</a>.</p> <p>This bit is Set when the Secondary Side for <a href="#">Type 1 Configuration Space Header</a> Function (for Requests completed by the Type 1 header Function itself) completes a Posted or Non-Posted Request as a Completer Abort error.</p> <p>Default value of this bit is 0b.</p>	<u>RW1C</u>
12	<p><b>Received Target Abort</b> - See <a href="#">Section 7.5.1.1.14</a>.</p> <p>This bit is Set when the Secondary Side for <a href="#">Type 1 Configuration Space Header</a> Function (for Requests initiated by the Type 1 header Function itself) receives a Completion with Completer Abort Completion Status.</p> <p>Default value of this bit is 0b.</p>	<u>RW1C</u>
13	<p><b>Received Master Abort</b> - See <a href="#">Section 7.5.1.1.14</a>.</p> <p>This bit is Set when the Secondary Side for <a href="#">Type 1 Configuration Space Header</a> Function (for Requests initiated by the Type 1 header Function itself) receives a Completion with Unsupported Request Completion Status.</p> <p>Default value of this bit is 0b.</p>	<u>RW1C</u>
14	<p><b>Received System Error</b> - See <a href="#">Section 7.5.1.1.14</a>.</p> <p>This bit is Set when the Secondary Side for a <a href="#">Type 1 Configuration Space Header</a> Function receives an <a href="#">ERR_FATAL</a> or <a href="#">ERR_NONFATAL</a> Message.</p> <p>Default value of this bit is 0b.</p>	<u>RW1C</u>
15	<p><b>Detected Parity Error</b> - See <a href="#">Section 7.5.1.1.14</a>.</p> <p>This bit is Set by a Function with a <a href="#">Type 1 Configuration Space Header</a> when a Poisoned TLP is received by its Secondary Side, regardless of the state the <a href="#">Parity Error Response Enable</a> bit in the <a href="#">Bridge Control Register</a>.</p> <p>Default value of this bit is 0b.</p>	<u>RW1C</u>

### **7.5.1.3.8 Memory Base Register/Memory Limit Register(Offset 20h/22h)**

The Memory Base and Memory Limit registers define a memory mapped address range which is used by the bridge to determine when to forward memory transactions from one interface to the other (see the [PCI-to-PCI-Bridge] for additional details).

The upper 12 bits of both the Memory Base and Memory Limit registers are read/write and correspond to the upper 12 address bits, Address[31:20], of 32-bit addresses. For the purpose of address decoding, the bridge assumes that the lower 20 address bits, Address[19:0], of the memory base address (not implemented in the Memory Base register) are zero. Similarly, the bridge assumes that the lower 20 address bits, Address[19:0], of the memory limit address (not implemented in the Memory Limit register) are F FFFFh. Thus, the bottom of the defined memory address range will be aligned to a 1 MB boundary and the top of the defined memory address range will be one less than a 1 MB boundary.

The Memory Limit register must be programmed to a smaller value than the Memory Base register if there is no memory-mapped address space on the secondary side of the bridge.

If there is no prefetchable memory space, and there is no memory-mapped space on the secondary side of the bridge, then the bridge will not forward any memory transactions from the primary bus to the secondary bus and will forward all memory transactions from the secondary bus to the primary bus.

The bottom four bits of both the Memory Base and Memory Limit registers are read-only and return zeros when read.

These registers must be initialized by configuration software so default states are not specified.

### **7.5.1.3.9 Prefetchable Memory Base/Prefetchable Memory Limit Registers (Offset 24h/26h)**

The Prefetchable Memory Base and Prefetchable Memory Limit registers must indicate that 64-bit addresses are supported.

The Prefetchable Memory Base and Prefetchable Memory Limit registers are optional. They define a prefetchable memory address range which is used by the bridge to determine when to forward memory transactions from one interface to the other.

If a bridge does not implement a prefetchable memory address range, then both Prefetchable Memory Base and Prefetchable Memory Limit registers must be implemented as read-only registers which return zero when read. If a bridge implements a Prefetchable memory address range, then both of these registers must be implemented as read/write registers. If a bridge supports a prefetchable memory address range, then these registers must be initialized by configuration software so default states are not specified.

If the bridge implements a prefetchable memory address range, the upper 12 bits of the register are read/write and correspond to the upper 12 address bits, Address[31:20], of 32-bit addresses. For the purpose of address decoding, the bridge assumes that the lower 20 address bits, Address[19:0], of the prefetchable memory base address (not implemented in the Prefetchable Memory Base register) are zero. Similarly, the bridge assumes that the lower 20 address bits, Address[19:0], of the prefetchable memory limit address (not implemented in the Prefetchable Memory Limit register) are F FFFFh. Thus, the bottom of the defined prefetchable memory address range will be aligned to a 1 MB boundary and the top of the defined memory address range will be one less than a 1 MB boundary.

The Prefetchable Memory Limit register must be programmed to a smaller value than the Prefetchable Memory Base register if there is no prefetchable memory on the secondary side of the bridge. If there is no prefetchable memory, and there is no memory-mapped address space (see the [PCI-to-PCI-Bridge]) on the secondary side of the bridge, then the bridge will not forward any memory transactions from the primary bus to the secondary but will forward all memory transactions from the secondary bus to the primary bus.

The bottom 4 bits of both the Prefetchable Memory Base and Prefetchable Memory Limit registers are read-only, contain the same value, and encode whether or not the bridge supports 64-bit addresses. If these four bits have the value 0h, then the bridge supports only 32-bit addresses. If these four bits have the value 01h, then the bridge supports 64-bit addresses and the Prefetchable Base Upper 32 Bits and Prefetchable Limit Upper 32 Bits registers hold the rest of the 64-bit prefetchable base and limit addresses respectively. All other encodings are Reserved.

### **7.5.1.3.10 Prefetchable Base Upper 32 Bits/Prefetchable Limit Upper 32 Bits Registers (Offset 28h/2Ch)**

The Prefetchable Base Upper 32 Bits and Prefetchable Limit Upper 32 Bits registers are optional extensions to the Prefetchable Memory Base and Prefetchable Memory Limit registers.

If the Prefetchable Memory Base and Prefetchable Memory Limit registers indicate support for 32-bit addressing, then the Prefetchable Base Upper 32 Bits and Prefetchable Limit Upper 32 Bits registers are both implemented as read-only registers that return zero when read. If the Prefetchable Memory Base and Prefetchable Memory Limit registers indicate support for 64-bit addressing, then the Prefetchable Base Upper 32 Bits and Prefetchable Limit Upper 32 Bits registers are implemented as read/write registers. If these registers are implemented as read/write registers, they must be initialized by configuration software so default states are not specified.

If a 64-bit prefetchable memory address range is supported, the Prefetchable Base Upper 32 Bits and Prefetchable Limit Upper 32 Bits registers specify the upper 32 bits, corresponding to Address[63:32], of the 64-bit base and limit addresses which specify the prefetchable memory address range (see the [PCI-to-PCI-Bridge] for additional details).

### **7.5.1.3.11 I/O Base Upper 16 Bits/I/O Limit Upper 16 Bits Registers (Offset 30h/32h)**

The I/O Base Upper 16 Bits and I/O Limit Upper 16 Bits registers are optional extensions to the I/O Base and I/O Limit registers. If the I/O Base and I/O Limit registers indicate support for 16-bit I/O address decoding, then the I/O Base Upper 16 Bits and I/O Limit Upper 16 Bits registers are implemented as read-only registers which return zero when read.

If the I/O Base and I/O Limit registers indicate support for 32-bit I/O addressing, then the I/O Base Upper 16 Bits and I/O Limit Upper 16 Bits registers must be initialized by configuration software so default states are not specified.

If 32-bit I/O address decoding is supported, the I/O Base Upper 16 Bits and the I/O Limit Upper 16 Bits registers specify the upper 16 bits, corresponding to Address[31:16], of the 32-bit base and limit addresses respectively, that specify the I/O address range (see the [PCI-to-PCI-Bridge] for additional details).

### **7.5.1.3.12 Expansion ROM Base Address Register (Offset 38h)**

This register is defined in Section 7.5.1.2.4. However the offset of the register within the Type 1 Configuration Space Header is different than that of the Type 0 Configuration Space Header.

### **7.5.1.3.13 Bridge Control Register (Offset 3Eh)**

The Bridge Control Register provides extensions to the Command Register that are specific to a Function with a Type 1 Configuration Space Header. The Bridge Control Register provides many of the same controls for the secondary interface that are provided by the Command Register for the primary interface. There are some bits that affect the operation of both interfaces of the bridge.

Table 7-12 defines the Bridge Control Register and Figure 7-16 depicts register layout. For PCI Express to PCI/PCI-X Bridges, refer to the [PCIe-to-PCI-PCI-X-Bridge] for requirements for this register.

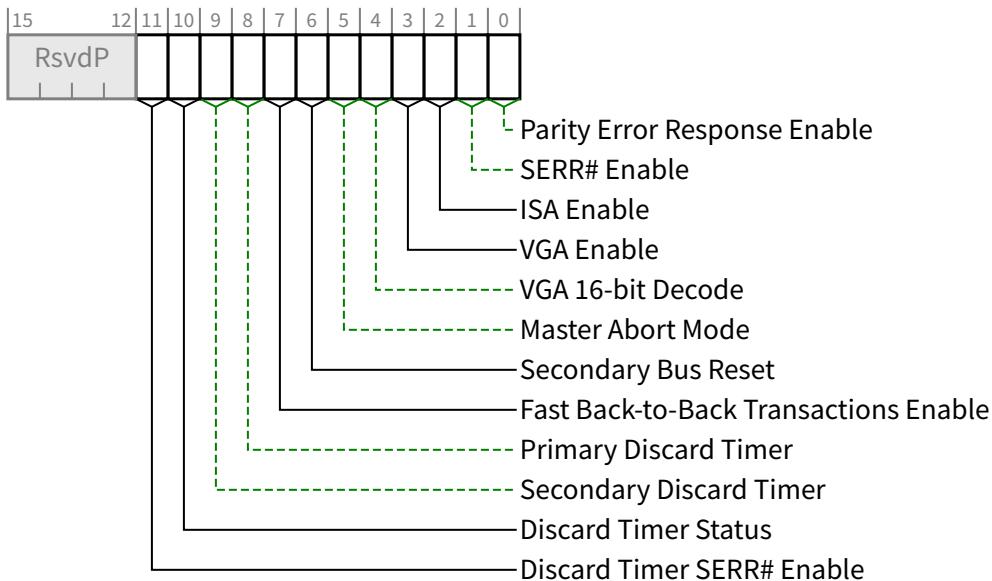


Figure 7-16 Bridge Control Register

Table 7-12 Bridge Control Register

Bit Location	Register Description	Attributes
0	<p><b>Parity Error Response Enable</b> - See Section 7.5.1.1.14 .</p> <p>This bit controls the logging of poisoned TLPs in the <u>Master Data Parity Error</u> bit in the <u>Secondary Status Register</u>.</p> <p>Default value of this bit is 0b.</p>	RW
1	<p><b>SERR# Enable</b> - See Section 7.5.1.1.14 .</p> <p>This bit controls forwarding of <u>ERR_COR</u>, <u>ERR_NONFATAL</u> and <u>ERR_FATAL</u> from secondary to primary.</p> <p>Default value of this bit is 0b.</p>	RW
2	<p><b>ISA Enable</b> - Modifies the response by the bridge to ISA I/O addresses. This applies only to I/O addresses that are enabled by the <u>I/O Base</u> and <u>I/O Limit</u> registers and are in the first 64 KB of I/O address space (0000 0000h to 0000 FFFFh). If this bit is Set, the bridge will block any forwarding from primary to secondary of I/O transactions addressing the last 768 bytes in each 1 KB block. In the opposite direction (secondary to primary), I/O transactions will be forwarded if they address the last 768 bytes in each 1 KB block.</p> <p><b>0b</b> forward downstream all I/O addresses in the address range defined by the <u>I/O Base</u> and <u>I/O Limit</u> registers</p> <p><b>1b</b> forward upstream ISA I/O addresses in the address range defined by the <u>I/O Base</u> and <u>I/O Limit</u> registers that are in the first 64 KB of PCI I/O address space (top 768 bytes of each 1 KB block)</p> <p>Default value of this bit is 0b.</p>	RW
3	<p><b>VGA Enable</b> - Modifies the response by the bridge to VGA compatible addresses. If the <u>VGA Enable</u> bit is Set, the bridge will positively decode and forward the following accesses on the primary interface to the</p>	RW

Bit Location	Register Description	Attributes
	<p>secondary interface (and, conversely, block the forwarding of these addresses from the secondary to primary interface):</p> <ul style="list-style-type: none"> <li>Memory accesses in the range 000A 0000h to 000B FFFFh</li> <li>I/O addresses in the first 64 KB of the I/O address space (Address[31:16] are 0000h) where Address[9:0] are in the ranges 3B0h to 3BBh and 3C0h to 3DFh (inclusive of ISA address aliases determined by the setting of VGA 16-bit Decode)</li> </ul> <p>If the <u>VGA Enable</u> bit is Set, forwarding of these accesses is independent of the I/O address range and memory address ranges defined by the <u>I/O Base</u> and <u>Limit</u> registers, the <u>Memory Base</u> and <u>Limit</u> registers, and the <u>Prefetchable Memory Base</u> and <u>Limit</u> registers of the bridge. (Forwarding of these accesses is also independent of the setting of the <u>ISA Enable</u> bit (in the <u>Bridge Control Register</u>) when the <u>VGA Enable</u> bit is Set. Forwarding of these accesses is qualified by the <u>I/O Space Enable</u> and <u>Memory Space Enable</u> bits in the <u>Command Register</u>.)</p> <p><b>0b</b> do not forward VGA compatible memory and I/O addresses from the primary to the secondary interface (addresses defined above) unless they are enabled for forwarding by the defined I/O and memory address ranges</p> <p><b>1b</b> forward VGA compatible memory and I/O addresses (addresses defined above) from the primary interface to the secondary interface (if the <u>I/O Space Enable</u> and <u>Memory Space Enable</u> bits are set) independent of the I/O and memory address ranges and independent of the <u>ISA Enable</u> bit</p> <p>Functions that do not support VGA are permitted to hardwire this bit to 0b.</p> <p>Default value of this bit is 0b.</p>	
4	<p><b>VGA 16-bit Decode</b> - This bit only has meaning if bit 3 (<u>VGA Enable</u>) of this register is also Set, enabling VGA I/O decoding and forwarding by the bridge.</p> <p>This bit enables system configuration software to select between 10-bit and 16-bit I/O address decoding for all VGA I/O register accesses that are forwarded from primary to secondary.</p> <p><b>0b</b> execute 10-bit address decodes on VGA I/O accesses</p> <p><b>1b</b> execute 16-bit address decodes on VGA I/O accesses</p> <p>Functions that do not support VGA are permitted to hardwire this bit to 0b.</p> <p>Default value of this bit is 0b.</p>	RW
5	<p><b>Master Abort Mode</b> - This bit was originally described in the [PCI-to-PCI-Bridge]. Its functionality does not apply to PCI Express and the bit must be hardwired to 0b.</p>	RO
6	<p><b>Secondary Bus Reset</b> - Setting this bit triggers a hot reset on the corresponding PCI Express Port. Software must ensure a minimum reset duration (<math>T_{rst}</math>). Software and systems must honor first-access-following-reset timing requirements defined in Section 6.6., unless the Readiness Notifications mechanism (see Section 6.23 ) is used or if the Immediate Readiness bit in the relevant Function's Status register is Set.</p> <p>Port configuration registers must not be changed, except as required to update Port status.</p> <p>Default value of this bit is 0b.</p>	RW
7	<p><b>Fast Back-to-Back Transactions Enable</b> - This bit was originally described in the [PCI-to-PCI-Bridge]. Its functionality does not apply to PCI Express and the bit must be hardwired to 0b.</p>	RO
8	<p><b>Primary Discard Timer</b> - This bit was originally described in the [PCI-to-PCI-Bridge]. Its functionality does not apply to PCI Express and the bit must be hardwired to 0b.</p>	RO
9	<p><b>Secondary Discard Timer</b> - This bit was originally described in the [PCI-to-PCI-Bridge]. Its functionality does not apply to PCI Express and the bit must be hardwired to 0b.</p>	RO

Bit Location	Register Description	Attributes
10	<b>Discard Timer Status</b> - This bit was originally described in the [PCI-to-PCI-Bridge]. Its functionality does not apply to PCI Express and the bit must be hardwired to 0b.	RO
11	<b>Discard Timer SERR# Enable</b> - This bit was originally described in the [PCI-to-PCI-Bridge]. Its functionality does not apply to PCI Express and must be hardwired to 0b.	RO

## 7.5.2 PCI Power Management Capability Structure

This section describes the registers making up the PCI Power Management Interface structure.

Figure 7-17 illustrates the organization of the PCI Power Management Capability structure. This structure is required for all PCI Express Functions.

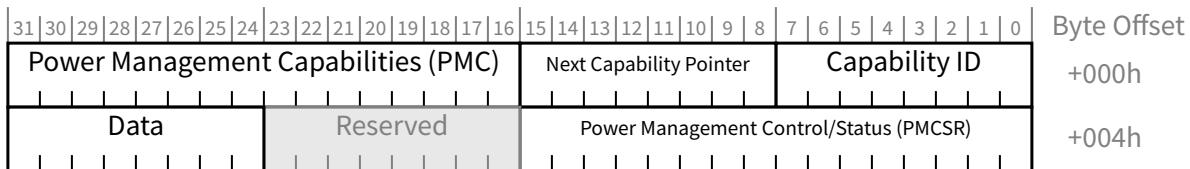


Figure 7-17 Power Management Capability Structure

Note: The 8-bit Data register (Offset 07h) is optional for both Type 0 and Type 1 Functions.

PCI Express device Functions are required to support D0 and D3 device states; PCI-PCI Bridge structures representing PCI Express Ports as described in Section 7.1 are required to indicate PME Message passing capability due to the in-band nature of PME messaging for PCI Express.

The PME\_Status bit for the PCI-PCI Bridge structure representing PCI Express Ports, however, is only Set when the PCI-PCI Bridge Function is itself generating a PME. The PME\_Status bit is not Set when the Bridge is propagating a PME Message but the PCI-PCI Bridge Function itself is not internally generating a PME.

### 7.5.2.1 Power Management Capabilities Register (Offset 00h)

Figure 7-18 details allocation of register fields for Power Management Capabilities register and Table 7-13 describes the requirements for this register.

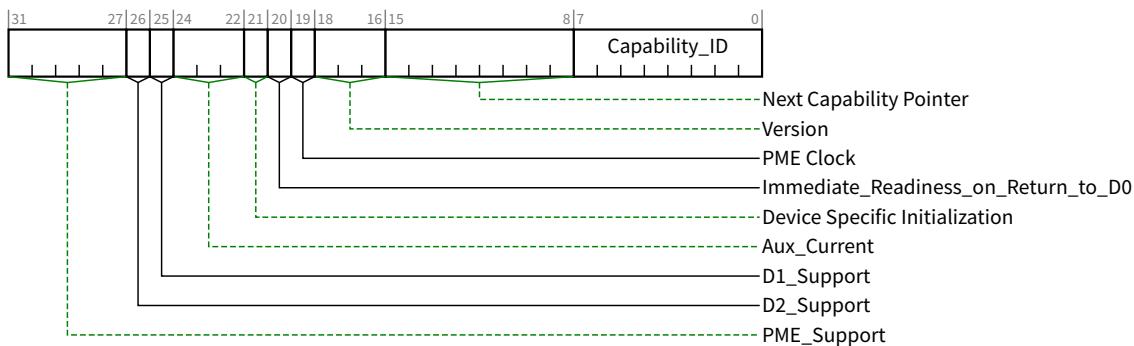


Figure 7-18 Power Management Capabilities Register

Table 7-13 Power Management Capabilities Register

Bit Location	Register Description	Attributes
7:0	<b>Capability_ID</b> - This field returns 01h to indicate that this is the PCI Power Management Capability. Each Function may have only one item in its capability list with <u>Capability_ID</u> set to 01h.	<u>RO</u>
15:8	<b>Next Capability Pointer</b> - This field provides an offset into the Function's Configuration Space pointing to the location of next item in the capabilities list. If there are no additional items in the capabilities list, this field is set to 00h.	<u>RO</u>
18:16	<b>Version</b> - Must be hardwired to 011b for Functions compliant to this specification.	<u>RO</u>
19	<b>PME Clock</b> - Does not apply to PCI Express and must be hardwired to 0b.	<u>RO</u>
20	<b>Immediate Readiness on Return to D0</b> - If this bit is a “1”, this Function is guaranteed to be ready to successfully complete valid accesses immediately after being set to <u>D0</u> . These accesses include Configuration cycles, and if the Function returns to <u>D0active</u> , they also include Memory and I/O Cycles. When this bit is “1”, for accesses to this Function, software is exempt from all requirements to delay accesses following a transition to <u>D0</u> , including but not limited to the 10 ms delay; the delays described in <u>Section 5.9</u> . How this guarantee is established is beyond the scope of this document. It is permitted that system software/firmware provide mechanisms that supersede the indication provided by this bit, however such software/firmware mechanisms are outside the scope of this specification.	<u>RO</u>
21	<b>Device Specific Initialization</b> - The DSI bit indicates whether special initialization of this Function is required. When Set indicates that the Function requires a device specific initialization sequence following a transition to the <u>D0uninitialized</u> state.	<u>RO</u>
24:22	<b>Aux_Current</b> - This 3 bit field reports the Vaux auxiliary current requirements for the Function. If this Function implements the Data Register, this field must be hardwired to 000b. If <u>PME_Support</u> is 0 xxxx b (PME assertion from <u>D3Cold</u> is not supported), this field must be hardwired to 0000b. For Functions where <u>PME_Support</u> is 1 xxxx b (PME assertion from <u>D3Cold</u> is supported), and which do not implement the Data register, the following encodings apply :	<u>RO</u>

Bit Location	Register Description	Attributes																
	<p>Encoding Vaux Max. Current Required</p> <table> <tr><td><b>111b</b></td><td>375 mA</td></tr> <tr><td><b>110b</b></td><td>320 mA</td></tr> <tr><td><b>101b</b></td><td>270 mA</td></tr> <tr><td><b>100b</b></td><td>220 mA</td></tr> <tr><td><b>011b</b></td><td>160 mA</td></tr> <tr><td><b>010b</b></td><td>100 mA</td></tr> <tr><td><b>001b</b></td><td>55 mA</td></tr> <tr><td><b>000b</b></td><td>0 (self powered)</td></tr> </table>	<b>111b</b>	375 mA	<b>110b</b>	320 mA	<b>101b</b>	270 mA	<b>100b</b>	220 mA	<b>011b</b>	160 mA	<b>010b</b>	100 mA	<b>001b</b>	55 mA	<b>000b</b>	0 (self powered)	
<b>111b</b>	375 mA																	
<b>110b</b>	320 mA																	
<b>101b</b>	270 mA																	
<b>100b</b>	220 mA																	
<b>011b</b>	160 mA																	
<b>010b</b>	100 mA																	
<b>001b</b>	55 mA																	
<b>000b</b>	0 (self powered)																	
25	<p><b>D1_Support</b> - If this bit is Set, this Function supports the <u>D1</u> Power Management State. Functions that do not support <u>D1</u> must always return a value of 0b for this bit.</p>	<u>RO</u>																
26	<p><b>D2_Support</b> - If this bit is Set, this Function supports the <u>D2</u> Power Management State. Functions that do not support <u>D2</u> must always return a value of 0b for this bit.</p>	<u>RO</u>																
31:27	<p><b>PME_Support</b> - This 5-bit field indicates the power states in which the Function may generate a PME and/or forward PME Messages. A value of 0b for any bit indicates that the Function is not capable of asserting PME while in that power state.</p> <table> <tr><td><b>bit(27) X XXX1b</b></td><td>PME can be generated from D0</td></tr> <tr><td><b>bit(28) X XX1Xb</b></td><td>PME can be generated from D1</td></tr> <tr><td><b>bit(29) X X1XXb</b></td><td>PME can be generated from D2</td></tr> <tr><td><b>bit(30) X 1XXXb</b></td><td>PME can be generated from <u>D3Hot</u></td></tr> <tr><td><b>bit(31) 1 XXXXb</b></td><td>PME can be generated from <u>D3Cold</u></td></tr> </table> <p>Bit 31 (PME can be asserted from <u>D3Cold</u>) represents a special case. Functions that Set this bit require some sort of auxiliary power source. Implementation specific mechanisms are recommended to validate that the power source is available before setting this bit.</p> <p>Each bit that corresponds to a supported D-state must be Set for PCI-PCI Bridge structures representing Ports on Root Complexes/Switches to indicate that the Bridge will forward PME Messages. Bit 31 must only be Set if the Port is still able to forward PME Messages when main power is not available.</p>	<b>bit(27) X XXX1b</b>	PME can be generated from D0	<b>bit(28) X XX1Xb</b>	PME can be generated from D1	<b>bit(29) X X1XXb</b>	PME can be generated from D2	<b>bit(30) X 1XXXb</b>	PME can be generated from <u>D3Hot</u>	<b>bit(31) 1 XXXXb</b>	PME can be generated from <u>D3Cold</u>	<u>RO</u>						
<b>bit(27) X XXX1b</b>	PME can be generated from D0																	
<b>bit(28) X XX1Xb</b>	PME can be generated from D1																	
<b>bit(29) X X1XXb</b>	PME can be generated from D2																	
<b>bit(30) X 1XXXb</b>	PME can be generated from <u>D3Hot</u>																	
<b>bit(31) 1 XXXXb</b>	PME can be generated from <u>D3Cold</u>																	

### 7.5.2.2 Power Management Control/Status Register (Offset 04h)

This register is used to manage the PCI Function's power management state as well as to enable/monitor PMEs.

The PME Context includes the value of the PME\_Status and PME\_En bits, implementation specific state needed during D3Cold to implement the wakeup functionality (e.g., recognized a Wake on LAN packet and generate a PME Message), as well as any additional implementation specific state that must be preserved during a transition to the D0uninitialized state.

If a Function supports PME generation from D3Cold, its PME Context is not affected by Reset. This is because the Function's PME functionality itself may have been responsible for the wake event which caused the transition back to D0. Therefore, the PME Context must be preserved for the system software to process.

If PME generation is not supported from D3Cold, then all PME Context is initialized with the assertion of Reset.

Figure 7-19 details allocation of the register fields for the Power Management Control/Status register and Table 7-14 describes the requirements for this register.

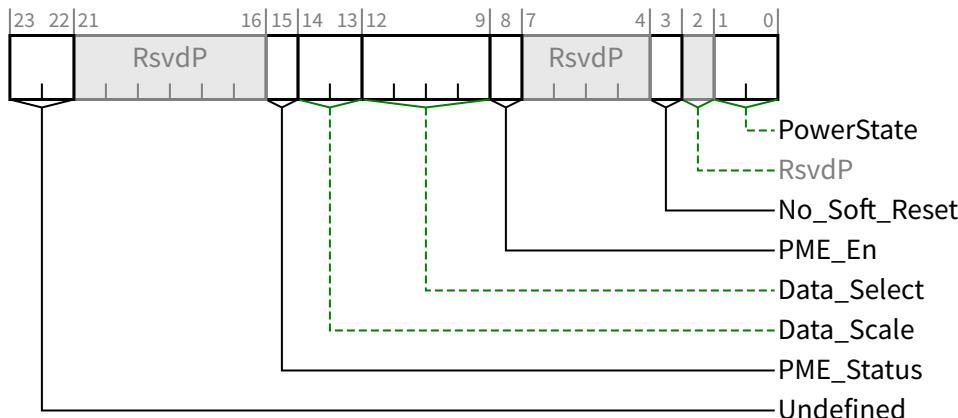


Figure 7-19 Power Management Control/Status Register

Table 7-14 Power Management Control/Status Register

Bit Location	Register Description	Attributes								
1:0	<p><b>PowerState</b> - This 2-bit field is used both to determine the current power state of a Function and to set the Function into a new power state. The definition of the field values is given below.</p> <table> <tr> <td><b>00b</b></td><td>D0</td></tr> <tr> <td><b>01b</b></td><td>D1</td></tr> <tr> <td><b>10b</b></td><td>D2</td></tr> <tr> <td><b>11b</b></td><td><u>D3Hot</u></td></tr> </table> <p>If software attempts to write an unsupported, optional state to this field, the write operation must complete normally; however, the data is discarded and no state change occurs.</p> <p>Default value of this field is 00b.</p>	<b>00b</b>	D0	<b>01b</b>	D1	<b>10b</b>	D2	<b>11b</b>	<u>D3Hot</u>	RW
<b>00b</b>	D0									
<b>01b</b>	D1									
<b>10b</b>	D2									
<b>11b</b>	<u>D3Hot</u>									
3	<p><b>No_Soft_Reset</b> - This bit indicates the state of the Function after writing the PowerState field to transition the Function from <u>D3Hot</u> to <u>D0</u>.</p> <p>When Set, this transition preserves internal Function state. The Function is in <u>D0Active</u> and no additional software intervention is required.</p> <p>When Clear, this transition results in undefined internal Function state.</p> <p>Regardless of this bit, Functions that transition from <u>D3Hot</u> to <u>D0</u> by Fundamental Reset will return to <u>D0uninitialized</u> with only PME context preserved if PME is supported and enabled.</p>	RO								
8	<p><b>PME_En</b> - When Set, the Function is permitted to generate a PME. When Clear, the Function is not permitted to generate a PME.</p> <p>If <u>PME_Support</u> is 1 xxxx b (PME generation from <u>D3Cold</u>) or the Function consumes auxiliary power and auxiliary power is available this bit is RWS and the bit is not modified by Conventional Reset or FLR</p> <p>If <u>PME_Support</u> is 0 xxxx b, this field is not sticky (RW).</p> <p>If <u>PME_Support</u> is 0 0000 b, this bit is permitted to be hardwired to 0b.</p>	RW/RWS								

Bit Location	Register Description	Attributes
12:9	<p><b>Data_Select</b> - This 4-bit field is used to select which data is to be reported through the Data register and Data_Scale field.</p> <p>If the Data register is not implemented, this field must be hardwired to 0000b.</p> <p>Refer to <a href="#">Section 7.5.2.3</a> for more details.</p> <p>The default of this field is 0000b</p>	<u>RW</u>
14:13	<p><b>Data_Scale</b> - This field indicates the scaling factor to be used when interpreting the value of the Data register. The value and meaning of this field will vary depending on which data value has been selected by the Data_Select field.</p> <p>This field is a required component of the Data register (offset 7) and must be implemented if the Data register is implemented.</p> <p>If the Data register is not implemented, this field must be hardwired to 00b.</p> <p>Refer to <a href="#">Section 7.5.2.3</a> for more details.</p>	<u>RO</u>
15	<p><b>PME_Status</b> - This bit is Set when the Function would normally generate a PME signal. The value of this bit is not affected by the value of the <u>PME_En</u> bit.</p> <p>If <u>PME_Support</u> bit 31 of the Power Management Capabilities register is Clear, this bit is permitted to be hardwired to 0b.</p> <p>Functions that consume auxiliary power must preserve the value of this sticky register when auxiliary power is available. In such Functions, this register value is not modified by Conventional Reset or FLR.</p>	<u>RW1CS</u>
23:22	Undefined - these bits were defined in previous specifications. They should be ignored by software.	<u>RO</u>

### 7.5.2.3 Data (Offset 07h)

The Data register is an optional, 8-bit read-only register that provides a mechanism for the Function to report state dependent operating power consumed or dissipation.

If the Data register is implemented, then the Data\_Select and Data\_Scale fields must also be implemented. If this register is not implemented, it must be hardwired to 00h.

Software may check for the presence of the Data register by writing different values into the Data\_Select field, looking for non-zero return data in the Data register and/or Data\_Scale field. Any non-zero Data register/Data\_Select read data indicates that the Data register complex has been implemented.

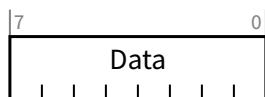


Figure 7-20 Data Register

*Table 7-15 Data Register*

Bit Location	Register Description	Attributes
7:0	<b>Data</b> - This register is used to report the state dependent data requested by the Data_Select field. The value of this register is scaled by the value reported by the Data_Scale field.	RO

The Data register is used by writing the proper value to the Data\_Select field in the PMCSR and then reading the Data\_Scale field and the Data register. The binary value read from Data is then multiplied by the scaling factor indicated by Data\_Scale to arrive at the value for the desired measurement. Table 7-16 shows which measurements are defined and how to interpret the values of each register.

*Table 7-16 Power Consumption/Dissipation Reporting*

Value in Data_Select	Data Reported	Data_Scale Interpretation	Units/Accuracy
0	D0 Power Consumed	0 = Unknown 1 = 0.1x 2 = 0.01x 3 = 0.001x	Watts
1	D1 Power Consumed		
2	D2 Power Consumed		
3	D3 Power Consumed		
4	D0 Power Dissipated		
5	D1 Power Dissipated		
6	D2 Power Dissipated		
7	D3 Power Dissipated		
8	Common logic power consumption ( <u>Multi-Function Devices</u> , Function 0 only)  <b>Function 0 of a Multi-Function Device:</b> Power consumption that is not associated with a specific Function.  <b>All other Functions:</b> Reserved		
9-15	Reserved	Reserved	TBD

The “Power Consumed” values defined above must include all power consumed from the power planes through the connector pins. If the add-in card provides power to external devices, that power must be included as well. It must not include any power derived from a battery or an external source. This information is useful for management of the power supply or battery.

The “Power Dissipated” values must provide the amount of heat which will be released into the interior of the computer chassis. This excludes any power delivered to external devices but must include any power derived from a battery or external power source and dissipated inside the computer chassis. This information is useful for fine grained thermal management.

Multi-Function Devices are recommended to report the power consumed by each Function in each corresponding Function’s Configuration Space. In a Multi-Function Device, power consumption for circuitry common to multiple Functions is reported in Function 0’s Configuration Space through the Data register once the Data\_Select field of Function 0’s Power Management Control/Status register has been programmed to 1000b. For a Multi-Function Device,

power consumption of the device is the sum of this value and, for every Function of the device, the reported value associated with the Function's current Power State.

Multiple component add-in cards implementing power reporting (i.e., multiple components behind a switch or bridge) must have the switch/bridge report the power it uses by itself. Each Function of each component on the add-in card is responsible for reporting the power consumed by that Function.

### 7.5.3 PCI Express Capability Structure

PCI Express defines a Capability structure in PCI-compatible Configuration Space (first 256 bytes) as shown in Figure 7-3 . This structure allows identification of a PCI Express device Function and indicates support for new PCI Express features. The PCI Express Capability structure is required for PCI Express device Functions. The Capability structure is a mechanism for enabling PCI software transparent features requiring support on legacy operating systems. In addition to identifying a PCI Express device Function, the PCI Express Capability structure is used to provide access to PCI Express specific Control/Status registers and related Power Management enhancements.

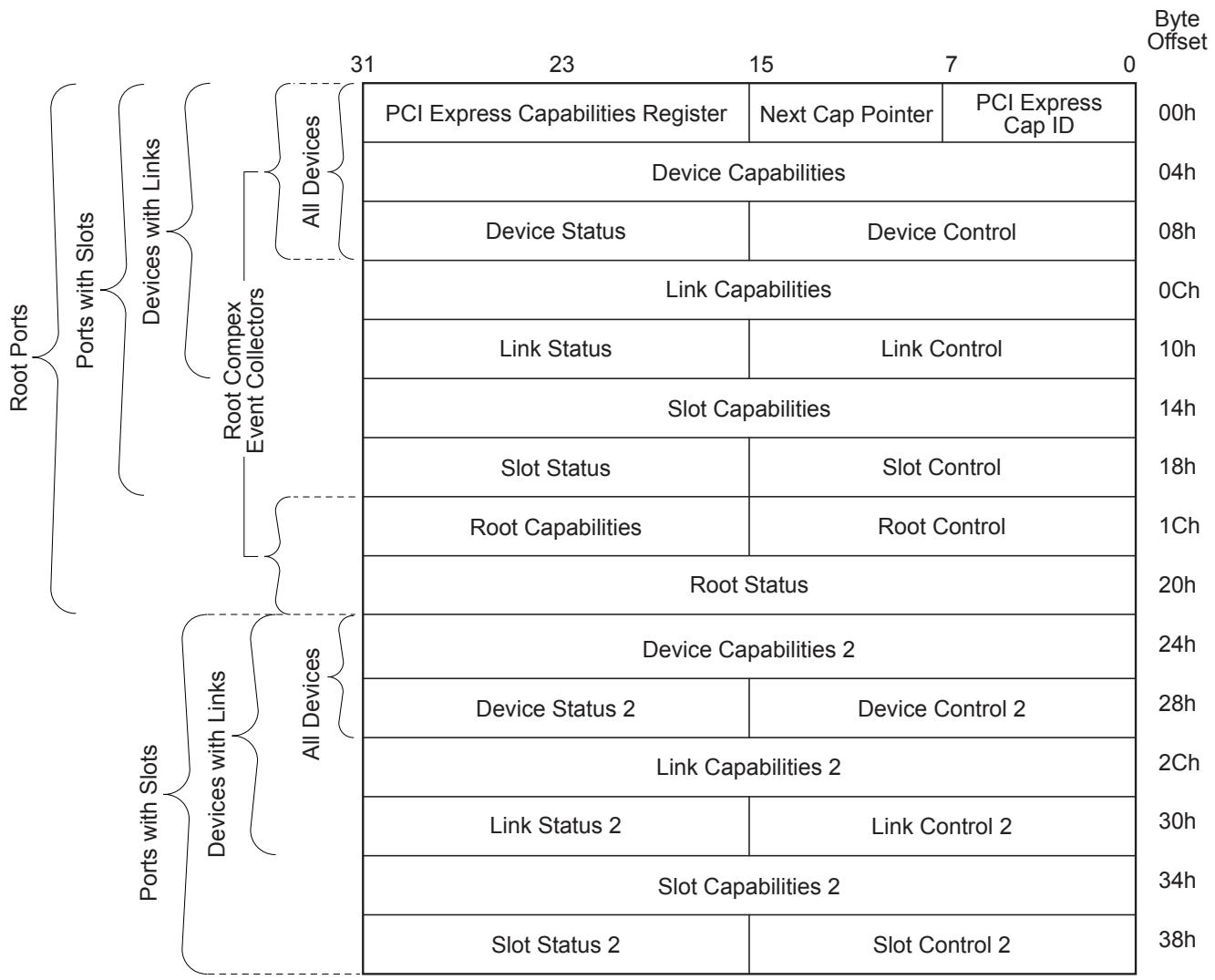
Figure 7-21 details allocation of register fields in the PCI Express Capability structure.

The PCI Express Capabilities, Device Capabilities, Device Status, and Device Control registers are required for all PCI Express device Functions. Device Capabilities 2, Device Status 2, and Device Control 2 registers are required for all PCI Express device Functions that implement capabilities requiring those registers. For device Functions that do not implement the Device Capabilities 2, Device Status 2, and Device Control 2 registers, these spaces must be hardwired to 0b.

The Link Capabilities, Link Status, and Link Control registers are required for all Root Ports, Switch Ports, Bridges, and Endpoints that are not RCiEPs. For Functions that do not implement the Link Capabilities, Link Status, and Link Control registers, these spaces must be hardwired to 0. Link Capabilities 2, Link Status 2, and Link Control 2 registers are required for all Root Ports, Switch Ports, Bridges, and Endpoints (except for RCiEPs) that implement capabilities requiring those registers. For Functions that do not implement the Link Capabilities 2, Link Status 2, and Link Control 2 registers, these spaces must be hardwired to 0b.

The Slot Capabilities, Slot Status, and Slot Control registers are required in certain Switch Downstream and Root Ports. The Slot Capabilities Register is required if the Slot Implemented bit is Set (see Section 7.5.3.2). The Slot Status and Slot Control registers are required if Slot Implemented is Set or if Data Link Layer Link Active Reporting Capable is Set (see Section 7.5.3.6). Switch Downstream and Root Ports are permitted to implement these registers, even when they are not required, and in this case the behavior of most of the fields in these registers is undefined. See Section 7.5.3.9, Section 7.5.3.10, and Section 7.5.3.11 for details. For Functions that do not implement the Slot Capabilities, Slot Status, and Slot Control registers, these spaces must be hardwired to 0b, with the exception of the Presence Detect State bit in the Slot Status Register of Downstream Ports, which must be hardwired to 1b (see Section 7.5.3.11). Slot Capabilities 2, Slot Status 2, and Slot Control 2 registers are required for Switch Downstream and Root Ports if the Function implements capabilities requiring those registers. For Functions that do not implement the Slot Capabilities 2, Slot Status 2, and Slot Control 2 registers, these spaces must be hardwired to 0b.

Root Ports and Root Complex Event Collectors must implement the Root Capabilities, Root Status, and Root Control registers. For Functions that do not implement the Root Capabilities, Root Status, and Root Control registers, these spaces must be hardwired to 0b.



Note: Registers not applicable to a device are RsvdZ.

OM14318B

Figure 7-21 *PCI Express Capability Structure*

### 7.5.3.1 PCI Express Capability List Register (Offset 00h)

The PCI Express Capability List Register enumerates the PCI Express Capability structure in the PCI Configuration Space Capability list. Figure 7-22 details allocation of register fields in the PCI Express Capability List Register; Table 7-17 provides the respective bit definitions.

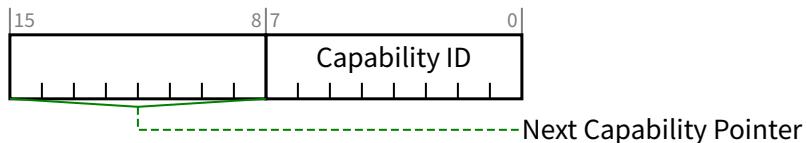


Figure 7-22 PCI Express Capability List Register

Table 7-17 PCI Express Capability List Register

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 10h indicating that this is a PCI Express Capability 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>

### 7.5.3.2 PCI Express Capabilities Register (Offset 02h)

The PCI Express Capabilities Register identifies PCI Express device Function type and associated capabilities. Figure 7-23 details allocation of register fields in the PCI Express Capabilities Register; Table 7-18 provides the respective bit definitions.

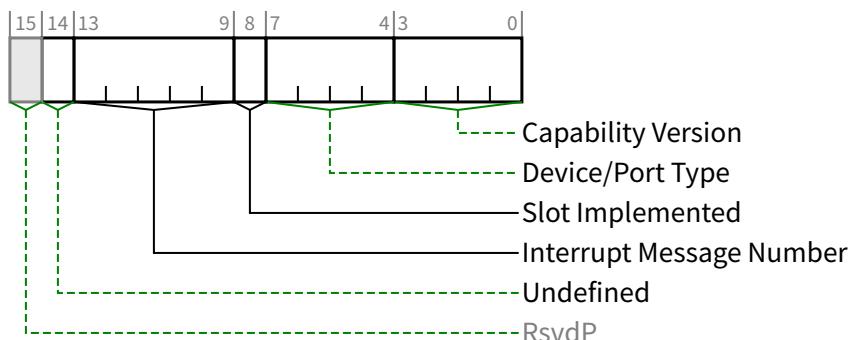


Figure 7-23 PCI Express Capabilities Register

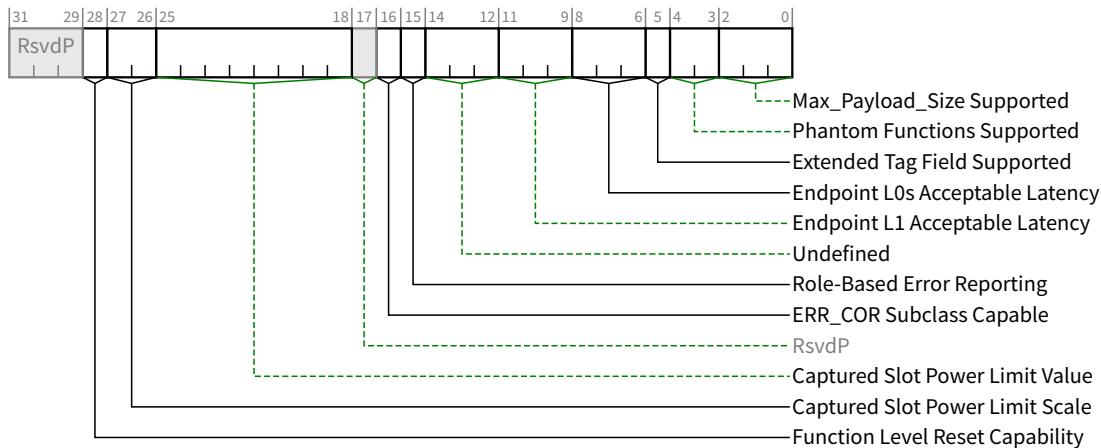
Table 7-18 PCI Express Capabilities Register

Bit Location	Register Description	Attributes
3:0	<b>Capability Version</b> - Indicates PCI-SIG defined PCI Express Capability structure version number. A version of the specification that changes the PCI Express Capability structure in a way that is not otherwise identifiable (e.g., through a new Capability field) is permitted to increment this field. All such changes to the PCI Express Capability structure must be software-compatible. Software must check for Capability Version numbers that are greater than or equal to the highest number defined when the	<u>RO</u>

Bit Location	Register Description	Attributes
	<p>software is written, as Functions reporting any such Capability Version numbers will contain a <u>PCI Express Capability</u> structure that is compatible with that piece of software.</p> <p>Must be hardwired to 2h for Functions compliant to this specification.</p>	
7:4	<p><b>Device/Port Type</b> - Indicates the specific type of this PCI Express Function. Note that different Functions in a <u>Multi-Function Device</u> can generally be of different types.</p> <p>Defined encodings for Functions that implement a Type 00h PCI Configuration Space header are:</p> <ul style="list-style-type: none"> <li><b>0000b</b> PCI Express Endpoint</li> <li><b>0001b</b> Legacy PCI Express Endpoint</li> <li><b>1001b</b> RCiEP</li> <li><b>1010b</b> Root Complex Event Collector</li> </ul> <p>Defined encodings for Functions that implement a Type 01h PCI Configuration Space header are:</p> <ul style="list-style-type: none"> <li><b>0100b</b> Root Port of PCI Express Root Complex</li> <li><b>0101b</b> Upstream Port of PCI Express Switch</li> <li><b>0110b</b> Downstream Port of PCI Express Switch</li> <li><b>0111b</b> PCI Express to PCI/PCI-X Bridge</li> <li><b>1000b</b> PCI/PCI-X to PCI Express Bridge</li> </ul> <p>All other encodings are Reserved.</p> <p>Note that the different Endpoint types have notably different requirements in <u>Section 1.3.2</u> regarding I/O resources, Extended Configuration Space, and other capabilities.</p>	RO
8	<p><b>Slot Implemented</b> - When Set, this bit indicates that the Link associated with this Port is connected to a slot (as compared to being connected to a system-integrated device or being disabled).</p> <p>This bit is valid for Downstream Ports. This bit is undefined for Upstream Ports.</p>	HwInit
13:9	<p><b>Interrupt Message Number</b> - This field indicates which MSI/MSI-X vector is used for the interrupt message generated in association with any of the status bits of this Capability structure.</p> <p>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 Multiple Message Enable field in the Message Control Register for MSI.</p> <p>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.</p> <p>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.</p>	RO
14	<p><b>Undefined</b> - The value read from this bit is undefined. In previous versions of this specification, this bit was used to indicate support for TCS Routing. System software should ignore the value read from this bit. System software is permitted to write any value to this bit.</p>	RO

### 7.5.3.3 Device Capabilities Register (Offset 04h)

The Device Capabilities Register identifies PCI Express device Function specific capabilities. Figure 7-24 details allocation of register fields in the Device Capabilities Register; Table 7-19 provides the respective bit definitions.



*Figure 7-24 Device Capabilities Register*

*Table 7-19 Device Capabilities Register*

Bit Location	Register Description	Attributes
2:0	<p><b>Max_Payload_Size Supported</b> - This field indicates the maximum payload size that the Function can support for TLPs.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>000b</b> 128 bytes max payload size</li> <li><b>001b</b> 256 bytes max payload size</li> <li><b>010b</b> 512 bytes max payload size</li> <li><b>011b</b> 1024 bytes max payload size</li> <li><b>100b</b> 2048 bytes max payload size</li> <li><b>101b</b> 4096 bytes max payload size</li> <li><b>110b</b> Reserved</li> <li><b>111b</b> Reserved</li> </ul> <p>The Functions of a <u>Multi-Function Device</u> are permitted to report different values for this field.</p>	RO
4:3	<p><b>Phantom Functions Supported</b> - This field indicates the support for use of unclaimed Function Numbers to extend the number of outstanding transactions allowed by logically combining unclaimed Function Numbers (called Phantom Functions) with the Tag identifier (see <u>Section 2.2.6.2</u> for a description of Tag Extensions).</p>	RO

Bit Location		Register Description	Attributes
	<p>With every Function in an <u>ARI Device</u>, the <u>Phantom Functions Supported</u> field must be set to 00b. The remainder of this field description applies only to non-ARI <u>Multi-Function Devices</u>.</p> <p>This field indicates the number of most significant bits of the Function Number portion of Requester ID that are logically combined with the Tag identifier.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>00b</b> No Function Number bits are used for Phantom Functions. <u>Multi-Function Devices</u> are permitted to implement up to 8 independent Functions.</li> <li><b>01b</b> The most significant bit of the Function number in Requester ID is used for Phantom Functions; a <u>Multi-Function Device</u> is permitted to implement Functions 0-3. Functions 0, 1, 2, and 3 are permitted to use Function Numbers 4, 5, 6, and 7 respectively as Phantom Functions.</li> <li><b>10b</b> The two most significant bits of Function Number in Requester ID are used for Phantom Functions; a <u>Multi-Function Device</u> is permitted to implement Functions 0-1. Function 0 is permitted to use Function Numbers 2, 4, and 6 for Phantom Functions. Function 1 is permitted to use Function Numbers 3, 5, and 7 as Phantom Functions.</li> <li><b>11b</b> All 3 bits of Function Number in Requester ID used for Phantom Functions. The device must have a single Function 0 that is permitted to use all other Function Numbers as Phantom Functions.</li> </ul> <p>Note that Phantom Function support for the Function must be enabled by the <u>Phantom Functions Enable</u> field in the <u>Device Control Register</u> before the Function is permitted to use the Function Number field in the Requester ID for Phantom Functions.</p>		
5	<p><b>Extended Tag Field Supported</b> - This bit, in combination with the <u>10-Bit Tag Requester Supported</u> bit in the <u>Device Capabilities 2 Register</u>, indicates the maximum supported size of the Tag field as a Requester. This bit must be Set if the <u>10-Bit Tag Requester Supported</u> bit is Set.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>0b</b> 5-bit Tag field supported</li> <li><b>1b</b> 8-bit Tag field supported</li> </ul> <p>Note that 8-bit Tag field generation must be enabled by the <u>Extended Tag Field Enable</u> bit in the <u>Device Control Register</u> of the Requester Function before 8-bit Tags can be generated by the Requester. See <u>Section 2.2.6.2</u> for interactions with enabling the use of 10-Bit Tags.</p>	RO	
8:6	<p><b>Endpoint L0s Acceptable Latency</b> - This field indicates the acceptable total latency that an Endpoint can withstand due to the transition from L0s state to the L0 state. It is essentially an indirect measure of the Endpoint's internal buffering.</p> <p>Power management software uses the reported L0s Acceptable Latency number to compare against the L0s exit latencies reported by all components comprising the data path from this Endpoint to the Root Complex Root Port to determine whether ASPM L0s entry can be used with no loss of performance.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>000b</b> Maximum of 64 ns</li> <li><b>001b</b> Maximum of 128 ns</li> <li><b>010b</b> Maximum of 256 ns</li> </ul>	RO	

Bit Location		Register Description	Attributes
	<p><b>011b</b> Maximum of 512 ns  <b>100b</b> Maximum of 1 <math>\mu</math>s  <b>101b</b> Maximum of 2 <math>\mu</math>s  <b>110b</b> Maximum of 4 <math>\mu</math>s  <b>111b</b> No limit</p> <p>For Functions other than Endpoints, this field is Reserved and must be hardwired to 000b.</p>		
11:9	<p><b>Endpoint L1 Acceptable Latency</b> - This field indicates the acceptable latency that an Endpoint can withstand due to the transition from L1 state to the L0 state. It is essentially an indirect measure of the Endpoint's internal buffering.</p> <p>Power management software uses the reported L1 Acceptable Latency number to compare against the L1 Exit Latencies reported (see below) by all components comprising the data path from this Endpoint to the Root Complex Root Port to determine whether ASPM L1 entry can be used with no loss of performance.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>000b</b> Maximum of 1 <math>\mu</math>s</li> <li><b>001b</b> Maximum of 2 <math>\mu</math>s</li> <li><b>010b</b> Maximum of 4 <math>\mu</math>s</li> <li><b>011b</b> Maximum of 8 <math>\mu</math>s</li> <li><b>100b</b> Maximum of 16 <math>\mu</math>s</li> <li><b>101b</b> Maximum of 32 <math>\mu</math>s</li> <li><b>110b</b> Maximum of 64 <math>\mu</math>s</li> <li><b>111b</b> No limit</li> </ul> <p>For Functions other than Endpoints, this field is Reserved and must be hardwired to 000b.</p>	RO	
14:12	<p><b>Undefined</b> - The value read from these bits are undefined. In previous versions of this specification, this bit was used to indicate that a Attention Button, Attention Indicator, or Power Indicator, is implemented on the adapter and electrically controlled by the component on the adapter. System software must ignore the value read from this bit. System software is permitted to write any value to this bit.</p>	RO	
15	<p><b>Role-Based Error Reporting</b> - When Set, this bit indicates that the Function implements the functionality originally defined in the Error Reporting ECN for [PCIe-1.0a], and later incorporated into [PCIe-1.1]. This bit must be Set by all Functions conforming to the ECN, [PCIe-1.1], or subsequent [PCIe] revisions.</p>	RO	
16	<p><b>ERR_COR Subclass Capable</b> - When Set, this bit indicates that the Function supports the ERR_COR Subclass field in ERR_COR Messages, allowing different subclasses to be distinguished. See <a href="#">Section 2.2.8.3</a>.</p> <p>Downstream Ports that implement the <u>System Firmware Intermediary (SFI)</u> capability must Set this bit. Downstream Ports that implement Downstream Port Containment (DPC) are strongly encouraged to Set this bit.</p>	RO	
25:18	<p><b>Captured Slot Power Limit Value</b> (Upstream Ports only) - In combination with the <u>Captured Slot Power Limit Scale</u> value, specifies the upper limit on power available to the adapter.</p> <p>Power limit (in Watts) is calculated by multiplying the value in this field by the value in the <u>Captured Slot Power Limit Scale</u> field except when the <u>Captured Slot Power Limit Scale</u></p>	RO	

Bit Location		Register Description	Attributes
	<p>field equals 00b (1.0x) and the <u>Captured Slot Power Limit Value</u> exceeds EFh, then alternative encodings are used (see <u>Section 7.5.3.9</u> ).</p> <p>This value is set by the <u>Set_Slot_Power_Limit Message</u> or hardwired to 00h (see <u>Section 6.9</u> ). The default value is 00h.</p>		
27:26	<p><b>Captured Slot Power Limit Scale</b> (Upstream Ports only) - Specifies the scale used for the <u>Slot Power Limit Value</u>.</p> <p>Range of Values:</p> <ul style="list-style-type: none"> <li><b>00b</b> 1.0x</li> <li><b>01b</b> 0.1x</li> <li><b>10b</b> 0.01x</li> <li><b>11b</b> 0.001x</li> </ul> <p>This value is set by the <u>Set_Slot_Power_Limit Message</u> or hardwired to 00b (see <u>Section 6.9</u> ). The default value is 00b.</p>		RO
28	<p><b>Function Level Reset Capability</b> - A value of 1b indicates the Function supports the optional Function Level Reset mechanism described in <u>Section 6.6.2</u> .</p> <p>This bit applies to Endpoints only. For all other Function types this bit must be hardwired to 0b.</p>		RO

### 7.5.3.4 Device Control Register (Offset 08h)

The Device Control Register controls PCI Express device specific parameters. Figure 7-25 details allocation of register fields in the Device Control Register; Table 7-20 provides the respective bit definitions.

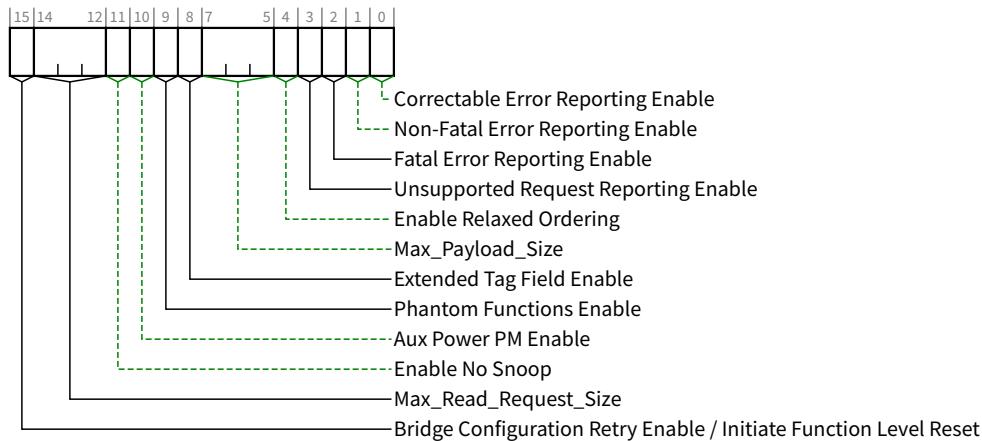


Figure 7-25 Device Control Register

Table 7-20 Device Control Register

Bit Location	Register Description	Attributes				
0	<p><b>Correctable Error Reporting Enable</b> - This bit, in conjunction with other bits, controls sending <u>ERR_COR</u> Messages (see <u>Section 6.2.5</u>, <u>Section 6.2.6</u>, and <u>Section 6.2.10.2</u> for details). For a <u>Multi-Function Device</u>, this bit controls error reporting for each Function from point-of-view of the respective Function.</p> <p>For a Root Port, the reporting of correctable errors is internal to the root. No external <u>ERR_COR</u> Message is generated.</p> <p>An <u>RCiEP</u> that is not associated with a Root Complex Event Collector is permitted to hardwire this bit to 0b.</p> <p>Default value of this bit is 0b.</p>	RW				
1	<p><b>Non-Fatal Error Reporting Enable</b> - This bit, in conjunction with other bits, controls sending <u>ERR_NONFATAL</u> Messages (see <u>Section 6.2.5</u> and <u>Section 6.2.6</u> for details). For a <u>Multi-Function Device</u>, this bit controls error reporting for each Function from point-of-view of the respective Function.</p> <p>For a Root Port, the reporting of Non-fatal errors is internal to the root. No external <u>ERR_NONFATAL</u> Message is generated.</p> <p>An <u>RCiEP</u> that is not associated with a Root Complex Event Collector is permitted to hardwire this bit to 0b.</p> <p>Default value of this bit is 0b.</p>	RW				
2	<p><b>Fatal Error Reporting Enable</b> - This bit, in conjunction with other bits, controls sending <u>ERR_FATAL</u> Messages (see <u>Section 6.2.5</u> and <u>Section 6.2.6</u> for details). For a <u>Multi-Function Device</u>, this bit controls error reporting for each Function from point-of-view of the respective Function.</p> <p>For a Root Port, the reporting of Fatal errors is internal to the root. No external <u>ERR_FATAL</u> Message is generated.</p> <p>An <u>RCiEP</u> that is not associated with a Root Complex Event Collector is permitted to hardwire this bit to 0b.</p> <p>Default value of this bit is 0b.</p>	RW				
3	<p><b>Unsupported Request Reporting Enable</b> - This bit, in conjunction with other bits, controls the signaling of Unsupported Request Errors by sending error Messages (see <u>Section 6.2.5</u> and <u>Section 6.2.6</u> for details). For a <u>Multi-Function Device</u>, this bit controls error reporting for each Function from point-of-view of the respective Function.</p> <p>An <u>RCiEP</u> that is not associated with a Root Complex Event Collector is permitted to hardwire this bit to 0b.</p> <p>Default value of this bit is 0b.</p>	RW				
4	<p><b>Enable Relaxed Ordering</b> - If this bit is Set, the Function is permitted to set the <u>Relaxed Ordering</u> bit in the Attributes field of transactions it initiates that do not require strong write ordering (see <u>Section 2.2.6.4</u> and <u>Section 2.4</u> ).</p> <p>A Function is permitted to hardwire this bit to 0b if it never sets the <u>Relaxed Ordering</u> attribute in transactions it initiates as a Requester.</p> <p>When not hardwired to 0b, the default value of this bit is 1b.</p>	RW				
7:5	<p><b>Max_Payload_Size</b> - This field sets maximum TLP payload size for the Function. As a Receiver, the Function must handle TLPs as large as the set value. As a Transmitter, the Function must not generate TLPs exceeding the set value. Permissible values that can be programmed are indicated by the <u>Max_Payload_Size Supported</u> field in the <u>Device Capabilities Register</u> (see <u>Section 7.5.3.3</u> ).</p> <p>Defined encodings for this field are:</p> <table> <tr> <td><b>000b</b></td> <td>128 bytes max payload size</td> </tr> <tr> <td><b>001b</b></td> <td>256 bytes max payload size</td> </tr> </table>	<b>000b</b>	128 bytes max payload size	<b>001b</b>	256 bytes max payload size	RW
<b>000b</b>	128 bytes max payload size					
<b>001b</b>	256 bytes max payload size					

Bit Location	Register Description	Attributes
0	<p><b>010b</b> 512 bytes max payload size  <b>011b</b> 1024 bytes max payload size  <b>100b</b> 2048 bytes max payload size  <b>101b</b> 4096 bytes max payload size  <b>110b</b> Reserved  <b>111b</b> Reserved</p> <p>Functions that support only the 128-byte max payload size are permitted to hardwire this field to 000b.</p> <p>System software is not required to program the same value for this field for all the Functions of a Multi-Function Device. Refer to <a href="#">Section 2.2.2</a> for important guidance.</p> <p>For ARI Devices, <u>Max_Payload_Size</u> is determined solely by the setting in Function 0. The settings in the other Functions always return whatever value software programmed for each, but otherwise are ignored by the component.</p> <p>Default value of this field is 000b.</p>	
8	<p><b>Extended Tag Field Enable</b> - This bit, in combination with the <u>10-Bit Tag Requester Enable</u> bit in the <u>Device Control 2 Register</u>, determines how many Tag field bits a Requester is permitted to use.</p> <p>The following applies when the <u>10-Bit Tag Requester Enable</u> bit is Clear. If the <u>Extended Tag Field Enable</u> bit is Set, the Function is permitted to use an 8-bit Tag field as a Requester. If the bit is Clear, the Function is restricted to a 5-bit Tag field.</p> <p>See <a href="#">Section 2.2.6.2</a> for required behavior when the <u>10-Bit Tag Requester Enable</u> bit is Set.</p> <p>If software changes the value of the <u>Extended Tag Field Enable</u> bit while the Function has outstanding Non-Posted Requests, the result is undefined.</p> <p>Functions that do not implement this capability hardwire this bit to 0b.</p> <p>Default value of this bit is implementation specific.</p>	RW
9	<p><b>Phantom Functions Enable</b> - This bit, in combination with the <u>10-Bit Tag Requester Enable</u> bit in the <u>Device Control 2 Register</u>, determines how many outstanding Non-Posted Requests a Requester is permitted to generate. See <a href="#">Section 2.2.6.2</a> for complete details.</p> <p>When Set, this bit enables a Function to use unclaimed Functions as Phantom Functions to extend the number of outstanding transaction identifiers. If the bit is Clear, the Function is not allowed to use Phantom Functions.</p> <p>Software should not change the value of this bit while the Function has outstanding Non-Posted Requests; otherwise, the result is undefined.</p> <p>Functions that do not implement this capability hardwire this bit to 0b.</p> <p>Default value of this bit is 0b.</p>	RW
10	<p><b>Aux Power PM Enable</b> - When Set this bit, enables a Function to draw auxiliary power independent of PME Aux power. Functions that require auxiliary power on legacy operating systems should continue to indicate PME Aux power requirements. Auxiliary power is allocated as requested in the <u>Aux_Current</u> field of the <u>Power Management Capabilities</u> register (PMC), independent of the <u>PME_En</u> bit in the <u>Power Management Control/Status</u> register (PMCSR) (see <a href="#">Chapter 5</a>). For <u>Multi-Function Devices</u>, a component is allowed to draw auxiliary power if at least one of the Functions has this bit set.</p> <p>Note: Functions that consume auxiliary power must preserve the value of this sticky register when auxiliary power is available. In such Functions, this bit is not modified by Conventional Reset.</p> <p>Functions that do not implement this capability hardwire this bit to 0b.</p>	RWS
11	<p><b>Enable No Snoop</b> - If this bit is Set, the Function is permitted to Set the <u>No Snoop</u> bit in the Requester Attributes of transactions it initiates that do not require hardware enforced cache coherency (see <a href="#">Section 2.2.6.2</a>).</p>	RW

Bit Location	Register Description	Attributes																
	<p>Section 2.2.6.5 ). Note that setting this bit to 1b should not cause a Function to Set the <u>No Snoop</u> attribute on all transactions that it initiates. Even when this bit is Set, a Function is only permitted to Set the <u>No Snoop</u> attribute on a transaction when it can guarantee that the address of the transaction is not stored in any cache in the system.</p> <p>This bit is permitted to be hardwired to 0b if a Function would never Set the <u>No Snoop</u> attribute in transactions it initiates.</p> <p>Default value of this bit is 1b.</p>																	
14:12	<p><b>Max_Read_Request_Size</b> - This field sets the maximum Read Request size for the Function as a Requester. The Function must not generate Read Requests with a size exceeding the set value. Defined encodings for this field are:</p> <table> <tr><td><b>000b</b></td><td>128 bytes maximum Read Request size</td></tr> <tr><td><b>001b</b></td><td>256 bytes maximum Read Request size</td></tr> <tr><td><b>010b</b></td><td>512 bytes maximum Read Request size</td></tr> <tr><td><b>011b</b></td><td>1024 bytes maximum Read Request size</td></tr> <tr><td><b>100b</b></td><td>2048 bytes maximum Read Request size</td></tr> <tr><td><b>101b</b></td><td>4096 bytes maximum Read Request size</td></tr> <tr><td><b>110b</b></td><td>Reserved</td></tr> <tr><td><b>111b</b></td><td>Reserved</td></tr> </table> <p>Functions that do not generate Read Requests larger than 128 bytes and Functions that do not generate Read Requests on their own behalf are permitted to implement this field as Read Only (RO) with a value of 000b.</p> <p>Default value of this field is 010b.</p>	<b>000b</b>	128 bytes maximum Read Request size	<b>001b</b>	256 bytes maximum Read Request size	<b>010b</b>	512 bytes maximum Read Request size	<b>011b</b>	1024 bytes maximum Read Request size	<b>100b</b>	2048 bytes maximum Read Request size	<b>101b</b>	4096 bytes maximum Read Request size	<b>110b</b>	Reserved	<b>111b</b>	Reserved	RW
<b>000b</b>	128 bytes maximum Read Request size																	
<b>001b</b>	256 bytes maximum Read Request size																	
<b>010b</b>	512 bytes maximum Read Request size																	
<b>011b</b>	1024 bytes maximum Read Request size																	
<b>100b</b>	2048 bytes maximum Read Request size																	
<b>101b</b>	4096 bytes maximum Read Request size																	
<b>110b</b>	Reserved																	
<b>111b</b>	Reserved																	
15	<p><b>Bridge Configuration Retry Enable / Initiate Function Level Reset</b> - this bit has a different meaning based on Function type:</p> <ul style="list-style-type: none"> <li>• <b>PCI Express to PCI/PCI-X Bridges:</b> <p><b>Bridge Configuration Retry Enable</b> - When Set, this bit enables PCI Express to PCI/PCI-X bridges to return Configuration Request Retry Status (CRS) in response to Configuration Requests that target devices below the bridge. Refer to [PCIe-to-PCI-PCI-X-Bridge] for further details.</p> <p>Default value of this bit is 0b.</p> </li> <li>• <b>Endpoints with Function Level Reset Capability set to 1b:</b> <p><b>Initiate Function Level Reset</b> - A write of 1b initiates Function Level Reset to the Function. The value read by software from this bit is always 0b.</p> </li> <li>• <b>All others:</b> <p><b>Reserved</b> - Must hardwire the bit to 0b.</p> </li> </ul>	<p><i>PCI Express to PCI/PCI-X Bridges:</i></p> <p>RW</p> <p><i>FLR Capable Endpoints:</i></p> <p>RW</p> <p><i>All others:</i></p> <p>RsvdP</p>																

## IMPLEMENTATION NOTE

### Software UR Reporting Compatibility with 1.0a Devices

With 1.0a device Functions,<sup>143</sup> if the Unsupported Request Reporting Enable bit is Set, the Function when operating as a Completer will send an uncorrectable error Message (if enabled) when a UR error is detected. On platforms where an uncorrectable error Message is handled as a System Error, this will break PC-compatible Configuration Space probing, so software/firmware on such platforms may need to avoid setting the Unsupported Request Reporting Enable bit.

With device Functions implementing Role-Based Error Reporting, setting the Unsupported Request Reporting Enable bit will not interfere with PC-compatible Configuration Space probing, assuming that the severity for UR is left at its default of non-fatal. However, setting the Unsupported Request Reporting Enable bit will enable the Function to report UR errors<sup>144</sup> detected with posted Requests, helping avoid this case for potential silent data corruption.

On platforms where robust error handling and PC-compatible Configuration Space probing is required, it is suggested that software or firmware have the Unsupported Request Reporting Enable bit Set for Role-Based Error Reporting Functions, but clear for 1.0a Functions. Software or firmware can distinguish the two classes of Functions by examining the Role-Based Error Reporting bit in the Device Capabilities Register.

<sup>143</sup>. In this context, “1.0a devices” are devices that do not implement Role-Based Error Reporting.

<sup>144</sup>. With Role-Based Error Reporting devices, setting the SERR# Enable bit in the Command Register also implicitly enables UR reporting.

## IMPLEMENTATION NOTE

### Use of Max\_Payload\_Size

The Max\_Payload\_Size mechanism allows software to control the maximum payload in packets sent by Endpoints to balance latency versus bandwidth trade-offs, particularly for isochronous traffic.

If software chooses to program the Max\_Payload\_Size of various System Elements to non-default values, it must take care to ensure that each packet does not exceed the Max\_Payload\_Size parameter of any System Element along the packet's path. Otherwise, the packet will be rejected by the System Element whose Max\_Payload\_Size parameter is too small.

Discussion of specific algorithms used to configure Max\_Payload\_Size to meet this requirement is beyond the scope of this specification, but software should base its algorithm upon factors such as the following:

- the Max\_Payload\_Size capability of each System Element within a hierarchy
- awareness of when System Elements are added or removed through Hot-Plug operations
- knowing which System Elements send packets to each other, what type of traffic is carried, what type of transactions are used, or if packet sizes are constrained by other mechanisms

For the case of firmware that configures System Elements in preparation for running legacy operating system environments, the firmware may need to avoid programming a Max\_Payload\_Size above the default of 128 bytes, which is the minimum supported by Endpoints.

For example, if the operating system environment does not comprehend PCI Express, firmware probably should not program a non-default Max\_Payload\_Size for a hierarchy that supports Hot-Plug operations. Otherwise, if no software is present to manage Max\_Payload\_Size settings when a new element is added, improper operation may result. Note that a newly added element may not even support a Max\_Payload\_Size setting as large as the rest of the hierarchy, in which case software may need to deny enabling the new element or reduce the Max\_Payload\_Size settings of other elements.

## IMPLEMENTATION NOTE

### Use of Max\_Read\_Request\_Size

The Max\_Read\_Request\_Size mechanism allows improved control of bandwidth allocation in systems where Quality of Service (QoS) is important for the target applications. For example, an arbitration scheme based on counting Requests (and not the sizes of those Requests) provides imprecise bandwidth allocation when some Requesters use much larger sizes than others. The Max\_Read\_Request\_Size mechanism can be used to force more uniform allocation of bandwidth, by restricting the upper size of Read Requests.

#### 7.5.3.5 Device Status Register (Offset 0Ah)

The Device Status Register provides information about PCI Express device (Function) specific parameters. Figure 7-26 details allocation of register fields in the Device Status Register; Table 7-21 provides the respective bit definitions.

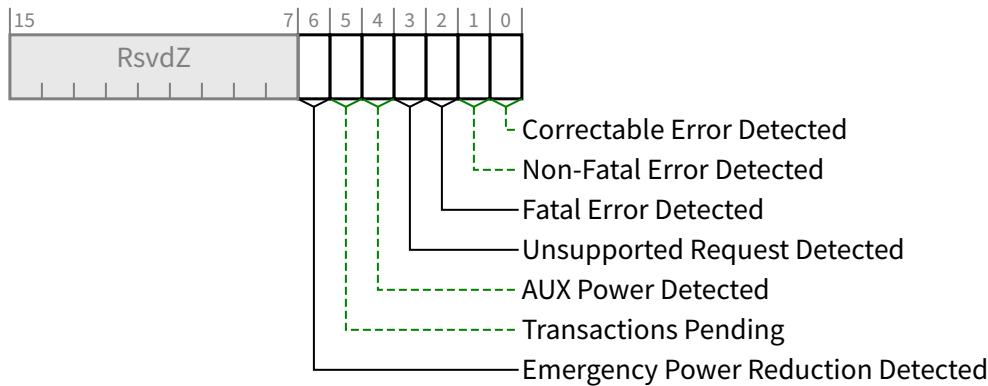


Figure 7-26 Device Status Register

Table 7-21 Device Status Register

Bit Location	Register Description	Attributes
0	<p><b>Correctable Error Detected</b> - This bit indicates status of correctable errors detected. Errors are logged in this register regardless of whether error reporting is enabled or not in the <u>Device Control Register</u>. For a Multi-Function Device, each Function indicates status of errors as perceived by the respective Function.</p> <p>For Functions supporting Advanced Error Handling, errors are logged in this register regardless of the settings of the Correctable Error Mask register.</p> <p>Default value of this bit is 0b.</p>	RW1C
1	<p><b>Non-Fatal Error Detected</b> - This bit indicates status of Non-fatal errors detected. Errors are logged in this register regardless of whether error reporting is enabled or not in the <u>Device Control Register</u>. For a Multi-Function Device, each Function indicates status of errors as perceived by the respective Function.</p> <p>For Functions supporting Advanced Error Handling, errors are logged in this register regardless of the settings of the Uncorrectable Error Mask register.</p> <p>Default value of this bit is 0b.</p>	RW1C
2	<p><b>Fatal Error Detected</b> - This bit indicates status of Fatal errors detected. Errors are logged in this register regardless of whether error reporting is enabled or not in the <u>Device Control Register</u>. For a Multi-Function Device, each Function indicates status of errors as perceived by the respective Function.</p> <p>For Functions supporting Advanced Error Handling, errors are logged in this register regardless of the settings of the Uncorrectable Error Mask register.</p> <p>Default value of this bit is 0b.</p>	RW1C
3	<p><b>Unsupported Request Detected</b> - This bit indicates that the Function received an Unsupported Request. Errors are logged in this register regardless of whether error reporting is enabled or not in the <u>Device Control Register</u>. For a Multi-Function Device, each Function indicates status of errors as perceived by the respective Function.</p> <p>Default value of this bit is 0b.</p>	RW1C
4	<b>AUX Power Detected</b> - Functions that require auxiliary power report this bit as Set if auxiliary power is detected by the Function.	RO
5	<b>Transactions Pending</b> -	RO

Bit Location	Register Description	Attributes
	<p><i>Endpoints:</i> When Set, this bit indicates that the Function has issued Non-Posted Requests that have not been completed. A Function reports this bit cleared only when all outstanding Non-Posted Requests have completed or have been terminated by the Completion Timeout mechanism. This bit must also be cleared upon the completion of an FLR.</p> <p><i>Root and Switch Ports:</i> When Set, this bit indicates that a Port has issued Non-Posted Requests on its own behalf (using the Port's own Requester ID) which have not been completed. The Port reports this bit cleared only when all such outstanding Non-Posted Requests have completed or have been terminated by the Completion Timeout mechanism. Note that Root and Switch Ports implementing only the functionality required by this document do not issue Non-Posted Requests on their own behalf, and therefore are not subject to this case. Root and Switch Ports that do not issue Non-Posted Requests on their own behalf hardwire this bit to 0b.</p>	
6	<p><b>Emergency Power Reduction Detected</b> - This bit is Set when the Function is in the Emergency Power Reduction State. Whenever any condition is present that would cause the Emergency Power Reduction State to be entered, the Function remains in the Emergency Power Reduction State and writes to this bit have no effect. See <a href="#">Section 6.25</a> for additional details.</p> <p>Multi-Function Devices associated with an Upstream Port must Set this bit in all Functions that support Emergency Power Reduction State.</p> <p>This bit is <a href="#">RsvdZ</a> if the <a href="#">Emergency Power Reduction Supported</a> field is 00b (see <a href="#">Section 7.5.3.15</a> ).</p> <p>This bit is <a href="#">RsvdZ</a> in Functions that are not associated with an Upstream Port.</p> <p>Default value is 0b.</p>	<a href="#">RW1C</a>

### 7.5.3.6 Link Capabilities Register (Offset 0Ch)

The [Link Capabilities Register](#) identifies PCI Express Link specific capabilities. [Figure 7-27](#) details allocation of register fields in the [Link Capabilities Register](#); [Table 7-22](#) provides the respective bit definitions.

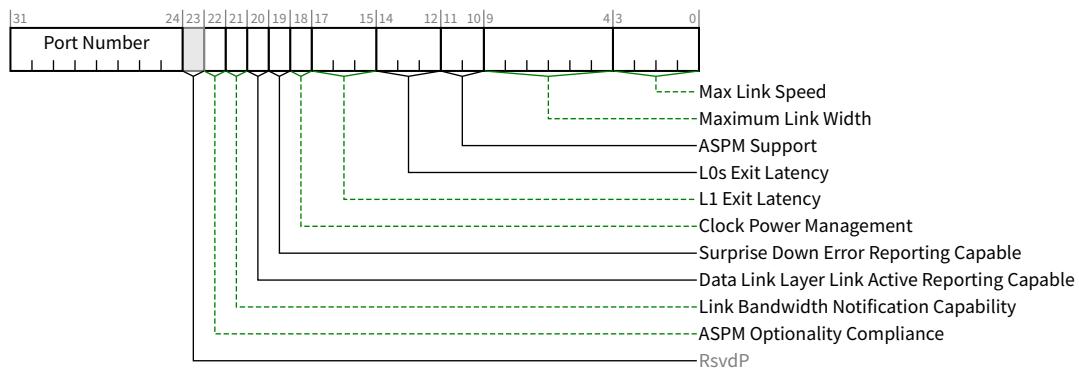


Figure 7-27 Link Capabilities Register

Table 7-22 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 Port. The encoded value specifies a Bit Location in the <u>Supported Link Speeds Vector</u> (in the <u>Link Capabilities 2 Register</u>) 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> </ul> <p>All other encodings are reserved.</p> <p>Multi-Function Devices associated with an Upstream Port must report the same value in this field for all Functions.</p>	RO
9:4	<p><b>Maximum Link Width</b> - This field indicates the maximum Link width (xN - corresponding to N Lanes) implemented by the component. This value is permitted to exceed the number of Lanes routed to the slot (Downstream Port), adapter connector (Upstream Port), or in the case of component-to-component connections, the actual wired connection width.</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.</p> <p>Multi-Function Devices associated with an Upstream Port must report the same value in this field for all Functions.</p>	RO
11:10	<p><b>ASPM Support / Active State Power Management Support</b> - This field indicates the level of ASPM supported on the given PCI Express Link. See <u>Section 5.4.1</u> for ASPM support requirements.</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> <li><b>10b</b> L1 Supported</li> <li><b>11b</b> L0s and L1 Supported</li> </ul> <p>Multi-Function Devices associated with an Upstream Port must report the same value in this field for all Functions.</p>	RO
14:12	<p><b>L0s Exit Latency</b> - This field indicates the L0s exit latency for the given PCI Express Link. The value reported indicates the length of time this Port requires to complete transition from L0s to L0. If L0s is not</p>	RO

Bit Location	Register Description	Attributes
	<p>supported, the value is undefined; however, see the Implementation Note “Potential Issues With Legacy Software When L0s is Not Supported” in <a href="#">Section 5.4.1.1</a> for the recommended value.</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 µs</li> <li><b>101b</b> 1 µs to less than 2 µs</li> <li><b>110b</b> 2 µs-4 µs</li> <li><b>111b</b> More than 4 µs</li> </ul> <p>Note that exit latencies may be influenced by PCI Express reference clock configuration depending upon whether a component uses a common or separate reference clock.</p> <p>Multi-Function Devices associated with an Upstream Port must report the same value in this field for all Functions.</p>	
17:15	<p><b>L1 Exit Latency</b> - This field indicates the L1 Exit Latency for the given PCI Express 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 1µs</li> <li><b>001b</b> 1 µs to less than 2 µs</li> <li><b>010b</b> 2 µs to less than 4 µs</li> <li><b>011b</b> 4 µs to less than 8 µs</li> <li><b>100b</b> 8 µs to less than 16 µs</li> <li><b>101b</b> 16 µs to less than 32 µs</li> <li><b>110b</b> 32 µs-64 µs</li> <li><b>111b</b> More than 64 µs</li> </ul> <p>Note that exit latencies may be influenced by PCI Express reference clock configuration depending upon whether a component uses a common or separate reference clock.</p> <p>Multi-Function Devices associated with an Upstream Port must report the same value in this field for all Functions.</p>	RO
18	<p><b>Clock Power Management</b> - For Upstream Ports, a value of 1b in this bit indicates that the component tolerates the removal of any reference clock(s) via the “clock request” (CLKREQ#) mechanism when the Link is in the L1 and L2/L3 Ready Link states. A value of 0b indicates the component does not have this capability and that reference clock(s) must not be removed in these Link states.</p> <p>L1 PM Substates defines other semantics for the CLKREQ# signal, which are managed independently of Clock Power Management.</p> <p>This Capability is applicable only in form factors that support “clock request” (CLKREQ#) capability.</p> <p>For a Multi-Function Device associated with an Upstream Port, each Function indicates its capability independently. Power Management configuration software must only permit reference clock removal if all Functions of the Multi-Function Device indicate a 1b in this bit. For ARI Devices, all Functions must indicate the same value in this bit.</p> <p>For Downstream Ports, this bit must be hardwired to 0b.</p>	RO

Bit Location	Register Description	Attributes
19	<p><b>Surprise Down Error Reporting Capable</b> - For a Downstream Port, this bit must be Set if the component supports the optional capability of detecting and reporting a Surprise Down error condition.</p> <p>For Upstream Ports and components that do not support this optional capability, this bit must be hardwired to 0b.</p>	<u>RO</u>
20	<p><b>Data Link Layer Link Active Reporting Capable</b> - For a Downstream Port, this bit must be hardwired to 1b if the component supports the optional capability of reporting the DL_Active state of the Data Link Control and Management State Machine. For a hot-plug capable Downstream Port (as indicated by the Hot-Plug Capable bit of the Slot Capabilities Register) or a Downstream Port that supports Link speeds greater than 5.0 GT/s, this bit must be hardwired to 1b.</p> <p>For Upstream Ports and components that do not support this optional capability, this bit must be hardwired to 0b.</p>	<u>RO</u>
21	<p><b>Link Bandwidth Notification Capability</b> - A value of 1b indicates support for the Link Bandwidth Notification status and interrupt mechanisms. This capability is required for all Root Ports and Switch Downstream Ports supporting Links wider than x1 and/or multiple Link speeds.</p> <p>This field is not applicable and is Reserved for Endpoints, PCI Express to PCI/PCI-X bridges, and Upstream Ports of Switches.</p> <p>Functions that do not implement the <u>Link Bandwidth Notification Capability</u> must hardwire this bit to 0b.</p>	<u>RO</u>
22	<p><b>ASPM Optionality Compliance</b> - This bit must be set to 1b in all Functions. Components implemented against certain earlier versions of this specification will have this bit set to 0b.</p> <p>Software is permitted to use the value of this bit to help determine whether to enable ASPM or whether to run ASPM compliance tests.</p>	<u>HwInit</u>
31:24	<p><b>Port Number</b> - This field indicates the PCI Express Port number for the given PCI Express Link.</p> <p>Multi-Function Devices associated with an Upstream Port must report the same value in this field for all Functions.</p>	<u>HwInit</u>

## IMPLEMENTATION NOTE

### Use of the ASPM Optionality Compliance Bit

Correct implementation and utilization of ASPM can significantly reduce Link power. However, ASPM feature implementations can be complex, and historically, some implementations have not been compliant to the specification. To address this, some of the ASPM optionality and ASPM entry requirements from earlier revisions of this document have been loosened. However, clear pass/fail compliance testing for ASPM features is also supported and expected.

The ASPM Optionality Compliance bit was created as a tool to establish clear expectations for hardware and software. This bit is Set to indicate hardware that conforms to the current specification, and this bit must be Set in components compliant to this specification.

System software as well as compliance software can assume that if this bit is Set, that the associated hardware conforms to the current specification. Hardware should be fully capable of supporting ASPM configuration management without needing component-specific treatment by system software.

For older hardware that does not have this bit Set, it is strongly recommended for system software to provide mechanisms to enable ASPM on components that work correctly with ASPM, and to disable ASPM on components that don't.

### 7.5.3.7 Link Control Register (Offset 10h)

The Link Control Register controls PCI Express Link specific parameters. Figure 7-28 details allocation of register fields in the Link Control Register; Table 7-23 provides the respective bit definitions.

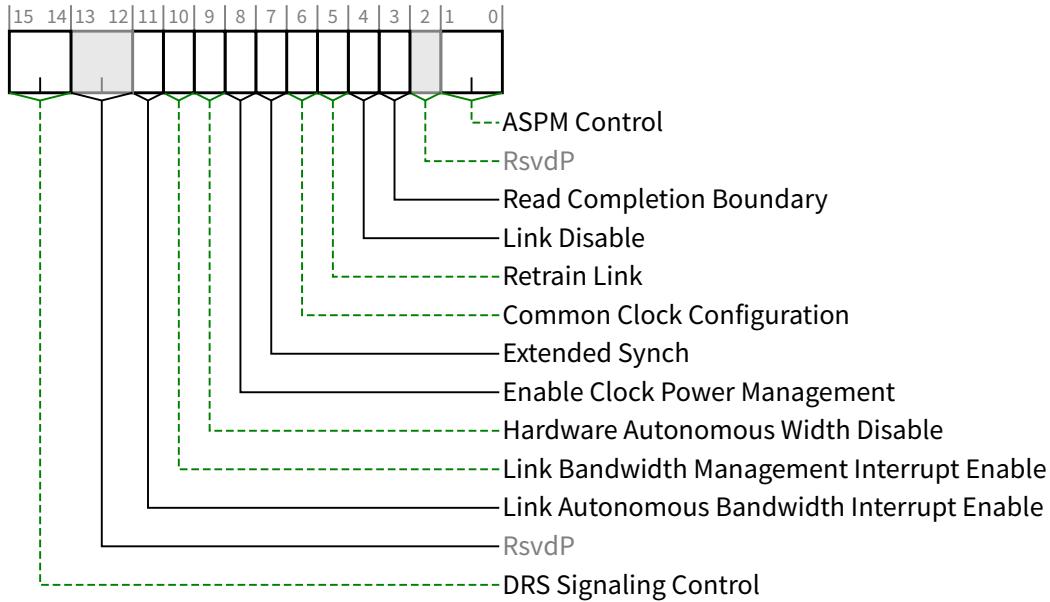


Figure 7-28 Link Control Register

Table 7-23 Link Control Register

Bit Location	Register Description	Attributes								
1:0	<p><b>ASPM Control / Active State Power Management Control</b> - This field controls the level of ASPM enabled on the given PCI Express Link. See <u>Section 5.4.1.3</u> for requirements on when and how to enable ASPM.</p> <p>Defined encodings are:</p> <table> <tr> <td><b>00b</b></td> <td>Disabled</td> </tr> <tr> <td><b>01b</b></td> <td>L0s Entry Enabled</td> </tr> <tr> <td><b>10b</b></td> <td>L1 Entry Enabled</td> </tr> <tr> <td><b>11b</b></td> <td>L0s and L1 Entry Enabled</td> </tr> </table> <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>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>	<b>00b</b>	Disabled	<b>01b</b>	L0s Entry Enabled	<b>10b</b>	L1 Entry Enabled	<b>11b</b>	L0s and L1 Entry Enabled	RW
<b>00b</b>	Disabled									
<b>01b</b>	L0s Entry Enabled									
<b>10b</b>	L1 Entry Enabled									
<b>11b</b>	L0s and L1 Entry Enabled									

Bit Location	Register Description	Attributes
	<p>For Multi-Function Devices (including ARI Devices), it is recommended that software program the same value for this field in all Functions. For non-ARI Multi-Function Devices, only capabilities enabled in all Functions are enabled for the component as a whole.</p> <p>For ARI Devices, ASPM Control is determined solely by the setting in Function 0, regardless of Function 0's D-state. The settings in the other Functions always return whatever value software programmed for each, but otherwise are ignored by the component.</p> <p>Default value of this field is 00b unless otherwise required by a particular form factor.</p>	
3	<p><b>Read Completion Boundary (RCB)</b> - field is meaningful in Root Ports, Endpoints and Bridges. When meaningful, defined encodings are:</p> <p><b>0b</b> 64 byte</p> <p><b>1b</b> 128 byte</p> <p><b>Root Ports:</b> RCB contains the the RCB value for the Root Port. Refer to Section 2.3.1.1 for the definition of the parameter RCB.</p> <p>This bit is hardwired for a Root Port and returns its RCB support capabilities.</p> <p><b>Endpoints and Bridges:</b> Read Completion Boundary (RCB) - Optionally Set by configuration software to indicate the RCB value of the Root Port Upstream from the Endpoint or Bridge. Refer to Section 2.3.1.1 for the definition of the parameter RCB.</p> <p>Configuration software must only Set this bit if the Root Port Upstream from the Endpoint or Bridge reports an RCB value of 128 bytes (a value of 1b in the Read Completion Boundary bit).</p> <p>Default value of this bit is 0b.</p> <p>Functions that do not implement this feature must hardwire the bit to 0b.</p> <p><b>Switch Ports:</b> Not applicable - must hardwire the bit to 0b</p>	<p><b>Root Ports:</b></p> <p>RO</p> <p><b>Endpoints and Bridges:</b></p> <p>RW</p> <p><b>Switch Ports:</b></p> <p>RO</p>
4	<p><b>Link Disable</b> - This bit disables the Link by directing the LTSSM to the Disabled state when Set; this bit is Reserved on Endpoints, PCI Express to PCI/PCI-X bridges, and Upstream Ports of Switches.</p> <p>Writes to this bit are immediately reflected in the value read from the bit, regardless of actual Link state.</p> <p>After clearing this bit, software must honor timing requirements defined in Section 6.6.1 with respect to the first Configuration Read following a Conventional Reset.</p> <p>Default value of this bit is 0b.</p>	RW
5	<p><b>Retrain Link</b> - A write of 1b to this bit initiates Link retraining by directing the Physical Layer LTSSM to the Recovery state. If the LTSSM is already in Recovery or Configuration, re-entering Recovery is permitted but not required. If the Port is in DPC when a write of 1b to this bit occurs, the result is undefined. Reads of this bit always return 0b.</p> <p>It is permitted to write 1b to this bit while simultaneously writing modified values to other fields in this register. If the LTSSM is not already in Recovery or Configuration, the resulting Link training must use the modified values. If the LTSSM is already in Recovery or Configuration, the modified values are not required to affect the Link training that's already in progress.</p> <p>This bit is not applicable and is Reserved for Endpoints, PCI Express to PCI/PCI-X bridges, and Upstream Ports of Switches.</p> <p>This bit always returns 0b when read.</p>	RW
6	<p><b>Common Clock Configuration</b> - When Set, this bit indicates that this component and the component at the opposite end of this Link are operating with a distributed common reference clock.</p>	RW

Bit Location	Register Description	Attributes
	<p>A value of 0b indicates that this component and the component at the opposite end of this Link are operating with asynchronous reference clock.</p> <p>For non-ARI Multi-Function Devices, software must program the same value for this bit in all Functions. If not all Functions are Set, then the component must as a whole assume that its reference clock is not common with the Upstream component.</p> <p>For ARI Devices, Common Clock Configuration is determined solely by the setting in Function 0. The settings in the other Functions always return whatever value software programmed for each, but otherwise are ignored by the component.</p> <p>Components utilize this Common Clock Configuration information to report the correct L0s and L1 Exit Latencies.</p> <p>After changing the value in this bit in both components on a Link, software must trigger the Link to retrain by writing a 1b to the Retrain Link bit of the Downstream Port.</p> <p>Default value of this bit is 0b.</p>	
7	<p><b>Extended Synch</b> - When Set, this bit 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>For Multi-Function Devices if any Function has this bit Set, then the component must transmit the additional Ordered Sets when exiting L0s or when in Recovery.</p> <p>Default value for this bit is 0b.</p>	RW
8	<p><b>Enable Clock Power Management</b> - Applicable only for Upstream Ports and with form factors that support a “Clock Request” (CLKREQ#) mechanism, this bit operates as follows:</p> <ul style="list-style-type: none"> <li><b>0b</b> Clock power management is disabled and device must hold CLKREQ# signal low.</li> <li><b>1b</b> When this bit is Set, the device is permitted to use CLKREQ# signal to power manage Link clock according to protocol defined in appropriate form factor specification.</li> </ul> <p>For a non-ARI Multi-Function Device, power-management-configuration software must only Set this bit if all Functions of the Multi-Function Device indicate a 1b in the Clock Power Management bit of the Link Capabilities Register. The component is permitted to use the CLKREQ# signal to power manage Link clock only if this bit is Set for all Functions.</p> <p>For ARI Devices, Clock Power Management is enabled solely by the setting in Function 0. The settings in the other Functions always return whatever value software programmed for each, but otherwise are ignored by the component.</p> <p>The CLKREQ# signal may also be controlled via the L1 PM Substates mechanism. Such control is not affected by the setting of this bit.</p> <p>Downstream Ports and components that do not support Clock Power Management (as indicated by a 0b value in the Clock Power Management bit of the Link Capabilities Register) must hardwire this bit to 0b.</p> <p>Default value of this bit is 0b, unless specified otherwise by the form factor specification.</p>	RW
9	<p><b>Hardware Autonomous Width Disable</b> - When Set, this bit disables hardware from changing the Link width for reasons other than attempting to correct unreliable Link operation by reducing Link width.</p> <p>For a Multi-Function Device associated with an Upstream Port, the bit in Function 0 is of type RW, and only Function 0 controls the component’s Link behavior. In all other Functions of that device, this bit is of type RsvdP.</p> <p>Components that do not implement the ability autonomously to change Link width are permitted to hardwire this bit to 0b.</p> <p>Default value of this bit is 0b.</p>	RW/RsvdP (see description)

Bit Location	Register Description	Attributes
10	<p><b>Link Bandwidth Management Interrupt Enable</b> - When Set, this bit enables the generation of an interrupt to indicate that the <u>Link Bandwidth Management Status</u> bit has been Set.</p> <p>This bit is not applicable and is Reserved for Endpoints, PCI Express-to-PCI/PCI-X bridges, and Upstream Ports of Switches.</p> <p>Functions that do not implement the <u>Link Bandwidth Notification Capability</u> must hardwire this bit to 0b.</p> <p>Default value of this bit is 0b.</p>	<u>RW</u>
11	<p><b>Link Autonomous Bandwidth Interrupt Enable</b> - When Set, this bit enables the generation of an interrupt to indicate that the Link Autonomous Bandwidth Status bit has been Set.</p> <p>This bit is not applicable and is Reserved for Endpoints, PCI Express-to-PCI/PCI-X bridges, and Upstream Ports of Switches.</p> <p>Functions that do not implement the <u>Link Bandwidth Notification Capability</u> must hardwire this bit to 0b.</p> <p>Default value of this bit is 0b.</p>	<u>RW</u>
15:14	<p><b>DRS Signaling Control</b> - Indicates the mechanism used to report reception of a DRS message. Must be implemented for Downstream Ports with the <u>DRS Supported</u> bit Set in the <u>Link Capabilities 2 Register</u>. Encodings are:</p> <ul style="list-style-type: none"> <li><b>00b</b> DRS not Reported: If <u>DRS Supported</u> is Set, receiving a DRS Message will set <u>DRS Message Received</u> in the <u>Link Status 2 Register</u> but will otherwise have no effect</li> <li><b>01b</b> DRS Interrupt Enabled: If the <u>DRS Message Received</u> bit in the <u>Link Status 2 Register</u> transitions from 0 to 1, and either MSI or MSI-X is enabled, an MSI or MSI-X interrupt is generated using the vector in Interrupt Message Number (Section 7.5.3.2.)</li> <li><b>10b</b> DRS to FRS Signaling Enabled: If the <u>DRS Message Received</u> bit in the <u>Link Status 2 Register</u> transitions from 0 to 1, the Port must send an FRS Message Upstream with the FRS Reason field set to DRS Message Received.</li> </ul> <p>Behavior is undefined if this field is set to 10b and the <u>FRS Supported</u> bit in the <u>Device Capabilities 2 Register</u> is Clear.</p> <p>Behavior is undefined if this field is set to 11b.</p> <p>Downstream Ports with the <u>DRS Supported</u> bit Clear in the <u>Link Capabilities 2 Register</u> must hardwire this field to 00b.</p> <p>This field is Reserved for Upstream Ports.</p> <p>Default value of this field is 00b.</p>	<u>RW/RsvdP</u>

## IMPLEMENTATION NOTE

### Software Compatibility with ARI Devices

With the ASPM Control field, Common Clock Configuration bit, and Enable Clock Power Management bit in the Link Control Register, there are potential software compatibility issues with ARI Devices since these controls operate strictly off the settings in Function 0 instead of the settings in all Functions.

With compliant software, there should be no issues with the Common Clock Configuration bit, since software is required to set this bit the same in all Functions.

With the Enable Clock Power Management bit, there should be no compatibility issues with software that sets this bit the same in all Functions. However, if software does not set this bit the same in all Functions, and relies on each Function having the ability to prevent Clock Power Management from being enabled, such software may have compatibility issues with ARI Devices.

With the ASPM Control field, there should be no compatibility issues with software that sets this bit the same in all Functions. However, if software does not set this bit the same in all Functions, and relies on each Function in D0 state having the ability to prevent ASPM from being enabled, such software may have compatibility issues with ARI Devices.

## IMPLEMENTATION NOTE

### Avoiding Race Conditions When Using the Retrain Link Bit

When software changes Link control parameters and writes a 1b to the Retain Link bit in order to initiate Link training using the new parameter settings, special care is required in order to avoid certain race conditions. At any instant the LTSSM may transition to the Recovery or Configuration state due to normal Link activity, without software awareness. If the LTSSM is already in Recovery or Configuration when software writes updated parameters to the Link Control Register, as well as a 1b, to the Retain Link bit, the LTSSM might not use the updated parameter settings with the current Link training, and the current Link training might not achieve the results that software intended.

To avoid this potential race condition, it is highly recommended that software use the following algorithm or something similar:

1. Software sets the relevant Link control parameters to the desired settings without writing a 1b to the Retain Link bit.
2. Software polls the Link Training bit in the Link Status Register until the value returned is 0b.
3. Software writes a 1b to the Retain Link bit without changing any other fields in the Link Control Register.

The above algorithm guarantees that Link training will be based on the Link control parameter settings that software intends.

## IMPLEMENTATION NOTE

### Use of the Slot Clock Configuration and Common Clock Configuration Bits

In order to determine the common clocking configuration of components on opposite ends of a Link that crosses a connector, two pieces of information are required. The following description defines these requirements.

The first necessary piece of information is whether the Port that connects to the slot uses a clock that has a common source and therefore constant phase relationship to the clock signal provided on the slot. This information is provided by the system side component through a hardware initialized bit (Slot Clock Configuration) in its Link Status Register. Note that some electromechanical form factor specifications may require the Port that connects to the slot use a clock that has a common source to the clock signal provided on the slot.

The second necessary piece of information is whether the component on the adapter uses the clock supplied on the slot or one generated locally on the adapter. The adapter design and layout will determine whether the component is connected to the clock source provided by the slot. A component going onto this adapter should have some hardware initialized method for the adapter design/designer to indicate the configuration used for this particular adapter design. This information is reported by bit 12 (Slot Clock Configuration) in the Link Status Register of each Function in the Upstream Port. Note that some electromechanical form factor specifications may require the Port on the adapter to use the clock signal provided on the connector.

System firmware or software will read this value from the components on both ends of a physical Link. If both components report the use of a common clock connection this firmware/software will program bit 6 (Common Clock Configuration) of the Link Control Register to 1b on both components connected to the Link. Each component uses this bit to determine the length of time required to re-synch its Receiver to the opposing component's Transmitter when exiting L0s.

This value is reported as a time value in bits 12-14 of the Link Capabilities Register (offset 0Ch) and is sent to the opposing Transmitter as part of the initialization process as N\_FTS. Components would be expected to require much longer sync times without common clocking and would therefore report a longer L0s Exit Latency in bits 12-14 of the Link Capabilities Register and would send a larger number for N\_FTS during training. This forces a requirement that whatever software changes this bit should force a Link retrain in order to get the correct N\_FTS set for the Receivers at both ends of the Link.

#### 7.5.3.8 Link Status Register (Offset 12h)

The Link Status Register provides information about PCI Express Link specific parameters. Figure 7-29 details allocation of register fields in the Link Status Register; Table 7-24 provides the respective bit definitions.

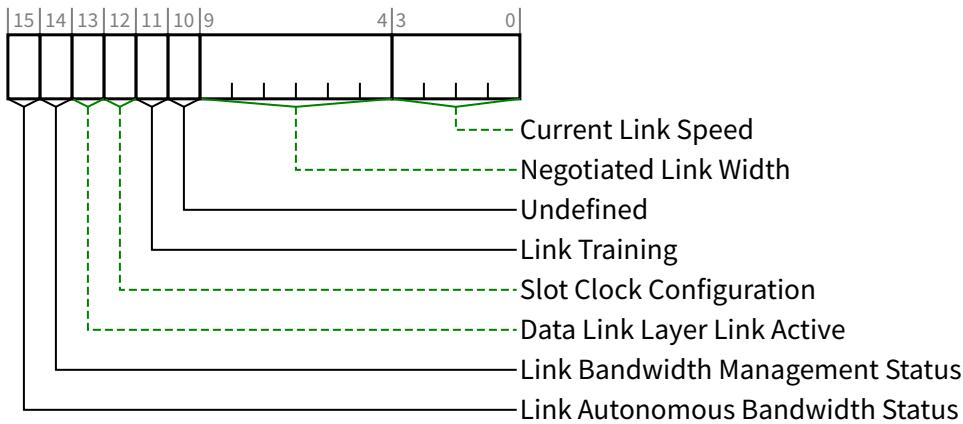


Figure 7-29 Link Status Register

Table 7-24 Link Status Register

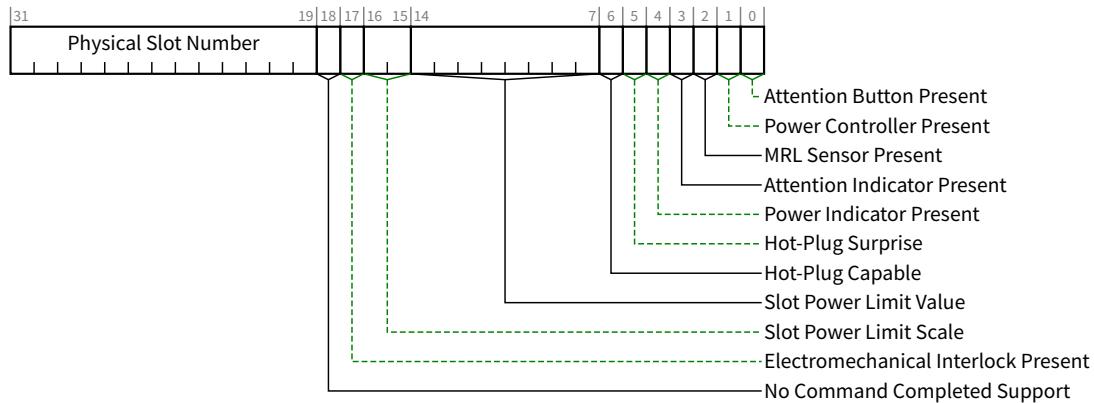
Bit Location	Register Description	Attributes
3:0	<p><b>Current Link Speed</b> - This field indicates the negotiated Link speed of the given PCI Express Link. The encoded value specifies a Bit Location in the <u>Supported Link Speeds Vector</u> (in the <u>Link Capabilities 2 Register</u>) that corresponds 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.</p>	<u>RO</u>
9:4	<p><b>Negotiated Link Width</b> - This field indicates the negotiated width of the given PCI Express 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.</p>	<u>RO</u>

Bit Location	Register Description	Attributes
10	<p><b>Undefined</b> - The value read from this bit is undefined. In previous versions of this specification, this bit was used to indicate a Link Training Error. System software must ignore the value read from this bit. System software is permitted to write any value to this bit.</p>	<u>RO</u>
11	<p><b>Link Training</b> - This read-only bit indicates that the Physical Layer LTSSM is in the Configuration or Recovery state, or that 1b was written to the <u>Retrain Link</u> bit but Link training has not yet begun. Hardware clears this bit when the LTSSM exits the Configuration/Recovery state.</p> <p>This bit is not applicable and Reserved for Endpoints, PCI Express to PCI/PCI-X bridges, and Upstream Ports of Switches, and must be hardwired to 0b.</p>	<u>RO</u>
12	<p><b>Slot Clock Configuration</b> - This bit indicates that the component uses the same physical reference clock that the platform provides on the connector. If the device uses an independent clock irrespective of the presence of a reference clock on the connector, this bit must be clear.</p> <p>For a <u>Multi-Function Device</u>, each Function must report the same value for this bit.</p>	<u>HwInit</u>
13	<p><b>Data Link Layer Link Active</b> - This bit indicates the status of the Data Link Control and Management State Machine. It returns a 1b to indicate the DL_Active state, 0b otherwise.</p> <p>This bit must be implemented if the <u>Data Link Layer Link Active Reporting Capable</u> bit is 1b. Otherwise, this bit must be hardwired to 0b.</p>	<u>RO</u>
14	<p><b>Link Bandwidth Management Status</b> - This bit is Set by hardware to indicate that either of the following has occurred without the Port transitioning through <u>DL_Down</u> status:</p> <ul style="list-style-type: none"> <li>• A Link retraining has completed following a write of 1b to the <u>Retrain Link</u> bit.</li> </ul> <p>Note: This bit is Set following any write of 1b to the <u>Retrain Link</u> bit, including when the Link is in the process of retraining for some other reason.</p> <ul style="list-style-type: none"> <li>• Hardware has changed Link speed or width to attempt to correct unreliable Link operation, either through an LTSSM timeout or a higher level process.</li> </ul> <p>This bit must be set if the Physical Layer reports a speed or width change was initiated by the Downstream component that was not indicated as an autonomous change.</p> <p>This bit is not applicable and is Reserved for Endpoints, PCI Express-to-PCI/PCI-X bridges, and Upstream Ports of Switches.</p> <p>Functions that do not implement the <u>Link Bandwidth Notification Capability</u> must hardwire this bit to 0b.</p> <p>The default value of this bit is 0b.</p>	<u>RW1C</u>
15	<p><b>Link Autonomous Bandwidth Status</b> - This bit is Set by hardware to indicate that hardware has autonomously changed Link speed or width, without the Port transitioning through <u>DL_Down</u> status, for reasons other than to attempt to correct unreliable Link operation.</p> <p>This bit must be set if the Physical Layer reports a speed or width change was initiated by the Downstream component that was indicated as an autonomous change.</p> <p>This bit is not applicable and is Reserved for Endpoints, PCI Express-to-PCI/PCI-X bridges, and Upstream Ports of Switches.</p> <p>Functions that do not implement the <u>Link Bandwidth Notification Capability</u> must hardwire this bit to 0b.</p> <p>The default value of this bit is 0b.</p>	<u>RW1C</u>

### 7.5.3.9 Slot Capabilities Register (Offset 14h)

The Slot Capabilities Register identifies PCI Express slot specific capabilities. Figure 7-30 details allocation of register fields in the Slot Capabilities Register; Table 7-25 provides the respective bit definitions.

If this register is implemented but the Slot Implemented bit is Clear, the field behavior of this entire register is undefined.



*Figure 7-30 Slot Capabilities Register*

*Table 7-25 Slot Capabilities Register*

Bit Location	Register Description	Attributes
0	<b>Attention Button Present</b> - When Set, this bit indicates that an Attention Button for this slot is electrically controlled by the chassis.	<u>HwInit</u>
1	<b>Power Controller Present</b> - When Set, this bit indicates that a software programmable Power Controller is implemented for this slot/adapter (depending on form factor).	<u>HwInit</u>
2	<b>MRL Sensor Present</b> - When Set, this bit indicates that an MRL Sensor is implemented on the chassis for this slot.	<u>HwInit</u>
3	<b>Attention Indicator Present</b> - When Set, this bit indicates that an Attention Indicator is electrically controlled by the chassis.	<u>HwInit</u>
4	<b>Power Indicator Present</b> - When Set, this bit indicates that a Power Indicator is electrically controlled by the chassis for this slot.	<u>HwInit</u>
5	<b>Hot-Plug Surprise</b> - When Set, this bit indicates that an adapter present in this slot might be removed from the system without any prior notification. This is a form factor specific capability. This bit is an indication to the operating system to allow for such removal without impacting continued software operation. If the <u>SFI HPS Suppress</u> bit in the <u>SFI Control Register</u> is Clear, a read of the <u>Hot-Plug Surprise</u> bit returns the <u>HwInit</u> value. If the <u>SFI HPS Suppress</u> bit is Set, a read returns 0b. See <u>Section 7.9.23.3</u> .	<u>HwInit/RO</u> (see description)
6	<b>Hot-Plug Capable</b> - When Set, this bit indicates that this slot is capable of supporting hot-plug operations.	<u>HwInit</u>

Bit Location	Register Description	Attributes
14:7	<p><b>Slot Power Limit Value</b> - In combination with the <u>Slot Power Limit Scale</u> value, specifies the upper limit on power supplied by the slot (see <u>Section 6.9</u>) or by other means to the adapter.</p> <p>Power limit (in Watts) is calculated by multiplying the value in this field by the value in the <u>Slot Power Limit Scale</u> field except when the <u>Slot Power Limit Scale</u> field equals 00b (1.0x) and <u>Slot Power Limit Value</u> exceeds EFh, the following alternative encodings are used:</p> <ul style="list-style-type: none"> <li><b>F0h</b> 250 W Slot Power Limit</li> <li><b>F1h</b> 275 W Slot Power Limit</li> <li><b>F2h</b> 300 W Slot Power Limit</li> <li><b>F3h to FFh</b> Reserved for Slot Power Limit Values above 300 W</li> </ul> <p>This register must be implemented if the <u>Slot Implemented</u> bit is Set.</p> <p>Writes to this register also cause the Port to send the <u>Set_Slot_Power_Limit Message</u>.</p> <p>The default value prior to hardware/firmware initialization is 0000 0000b.</p>	<u>HwInit</u>
16:15	<p><b>Slot Power Limit Scale</b> - Specifies the scale used for the <u>Slot Power Limit Value</u> (see <u>Section 6.9</u>).</p> <p>Range of Values:</p> <ul style="list-style-type: none"> <li><b>00b</b> 1.0x</li> <li><b>01b</b> 0.1x</li> <li><b>10b</b> 0.01x</li> <li><b>11b</b> 0.001x</li> </ul> <p>This register must be implemented if the <u>Slot Implemented</u> bit is Set.</p> <p>Writes to this register also cause the Port to send the <u>Set_Slot_Power_Limit Message</u>.</p> <p>The default value prior to hardware/firmware initialization is 00b.</p>	<u>HwInit</u>
17	<b>Electromechanical Interlock Present</b> - When Set, this bit indicates that an Electromechanical Interlock is implemented on the chassis for this slot.	<u>HwInit</u>
18	<b>No Command Completed Support</b> - When Set, this bit indicates that this slot does not generate software notification when an issued command is completed by the Hot-Plug Controller. This bit is only permitted to be Set if the hot-plug capable Port is able to accept writes to all fields of the <u>Slot Control Register</u> without delay between successive writes.	<u>HwInit</u>
31:19	<b>Physical Slot Number</b> - This field indicates the physical slot number attached to this Port. This field must be hardware initialized to a value that assigns a slot number that is unique within the chassis, regardless of the form factor associated with the slot. This field must be initialized to zero for Ports connected to devices that are either integrated on the system board or integrated within the same silicon as the Switch device or Root Port.	<u>HwInit</u>

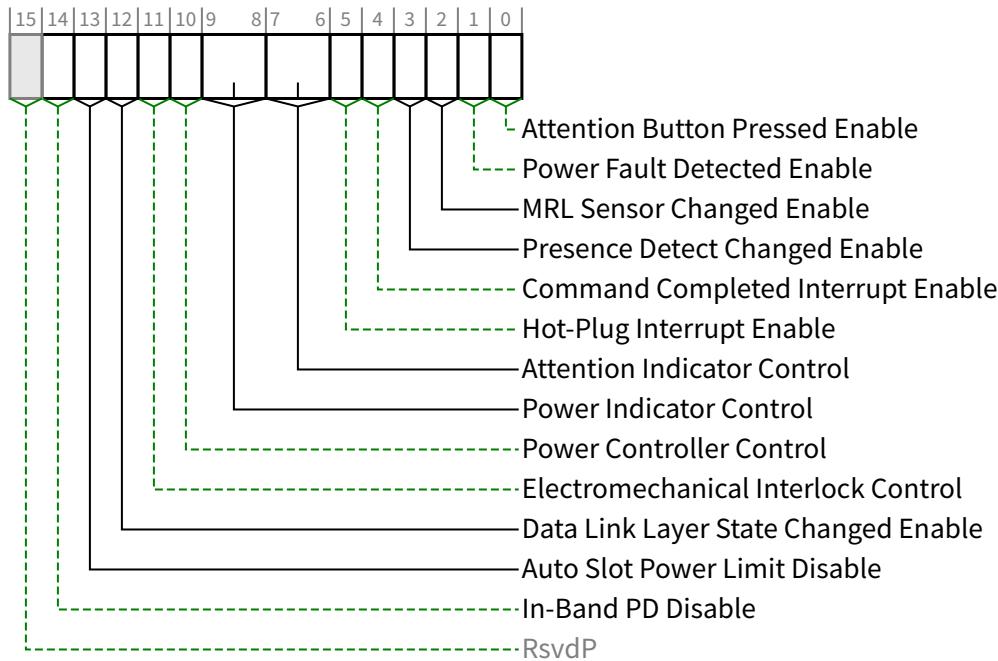
### 7.5.3.10 Slot Control Register (Offset 18h)

The Slot Control Register controls PCI Express Slot specific parameters. Figure 7-31 details allocation of register fields in the Slot Control Register; Table 7-26 provides the respective bit definitions.

Attention Indicator Control, Power Indicator Control, and Power Controller Control fields of the Slot Control Register do not have a defined default value. If these fields are implemented, it is the responsibility of either system firmware or operating system software to (re)initialize these fields after a reset of the Link.

In hot-plug capable Downstream Ports, a write to the Slot Control Register must cause a hot-plug command to be generated (see [Section 6.7.3.2](#) for details on hot-plug commands). A write to the Slot Control Register in a Downstream Port that is not hot-plug capable must not cause a hot-plug command to be executed.

If this register is implemented but the Slot Implemented bit is Clear, the field behavior of this entire register with the exception of the Data Link Layer State Changed Enable bit is undefined.



*Figure 7-31 Slot Control Register*

*Table 7-26 Slot Control Register*

Bit Location	Register Description	Attributes
0	<p><b>Attention Button Pressed Enable</b> - When Set to 1b, this bit enables software notification on an attention button pressed event (see <a href="#">Section 6.7.3</a>).</p> <p>If the <u>Attention Button Present</u> bit in the <u>Slot Capabilities 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</u>
1	<p><b>Power Fault Detected Enable</b> - When Set, this bit enables software notification on a power fault event (see <a href="#">Section 6.7.3</a>).</p> <p>If a Power Controller that supports power fault detection is not implemented, 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</u>
2	<p><b>MRL Sensor Changed Enable</b> - When Set, this bit enables software notification on a MRL sensor changed event (see <a href="#">Section 6.7.3</a>).</p> <p>If the <u>MRL Sensor Present</u> bit in the <u>Slot Capabilities Register</u> is Clear, this bit is permitted to be read-only with a value of 0b.</p>	<u>RW</u>

Bit Location	Register Description	Attributes								
	Default value of this bit is 0b.									
3	<p><b>Presence Detect Changed Enable</b> - When Set, this bit enables software notification on a presence detect changed event (see <a href="#">Section 6.7.3</a> ).</p> <p>If the <u>Hot-Plug Capable</u> bit in the <u>Slot Capabilities 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</u>								
4	<p><b>Command Completed Interrupt Enable</b> - If Command Completed notification is supported (if the <u>No Command Completed Support</u> bit in the <u>Slot Capabilities Register</u> is 0b), when Set, this bit enables software notification when a hot-plug command is completed by the Hot-Plug Controller.</p> <p>If <u>Command Completed</u> notification is not supported, this bit must be hardwired to 0b.</p> <p>Default value of this bit is 0b.</p>	<u>RW</u>								
5	<p><b>Hot-Plug Interrupt Enable</b> - When Set, this bit enables generation of an interrupt on enabled hot-plug events.</p> <p>If the <u>Hot-Plug Capable</u> bit in the <u>Slot Capabilities Register</u> is Clear, 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</u>								
7:6	<p><b>Attention Indicator Control</b> - If an Attention Indicator is implemented, writes to this field set the Attention Indicator to the written state.</p> <p>Reads of this field must reflect the value from the latest write, even if the corresponding hot-plug command is not complete, unless software issues a write without waiting, if required to, for the previous command to complete in which case the read value is undefined.</p> <p>Defined encodings are:</p> <table> <tr> <td><b>00b</b></td> <td>Reserved</td> </tr> <tr> <td><b>01b</b></td> <td>On</td> </tr> <tr> <td><b>10b</b></td> <td>Blink</td> </tr> <tr> <td><b>11b</b></td> <td>Off</td> </tr> </table> <p>Note: The default value of this field must be one of the non-Reserved values. If the <u>Attention Indicator Present</u> bit in the <u>Slot Capabilities Register</u> is 0b, this bit is permitted to be read-only with a value of 00b.</p>	<b>00b</b>	Reserved	<b>01b</b>	On	<b>10b</b>	Blink	<b>11b</b>	Off	<u>RW</u>
<b>00b</b>	Reserved									
<b>01b</b>	On									
<b>10b</b>	Blink									
<b>11b</b>	Off									
9:8	<p><b>Power Indicator Control</b> - If a Power Indicator is implemented, writes to this field set the Power Indicator to the written state. Reads of this field must reflect the value from the latest write, even if the corresponding hot-plug command is not complete, unless software issues a write without waiting, if required to, for the previous command to complete in which case the read value is undefined.</p> <p>Defined encodings are:</p> <table> <tr> <td><b>00b</b></td> <td>Reserved</td> </tr> <tr> <td><b>01b</b></td> <td>On</td> </tr> <tr> <td><b>10b</b></td> <td>Blink</td> </tr> <tr> <td><b>11b</b></td> <td>Off</td> </tr> </table> <p>Note: The default value of this field must be one of the non-Reserved values. If the <u>Power Indicator Present</u> bit in the <u>Slot Capabilities Register</u> is 0b, this bit is permitted to be read-only with a value of 00b.</p>	<b>00b</b>	Reserved	<b>01b</b>	On	<b>10b</b>	Blink	<b>11b</b>	Off	<u>RW</u>
<b>00b</b>	Reserved									
<b>01b</b>	On									
<b>10b</b>	Blink									
<b>11b</b>	Off									
10	<p><b>Power Controller Control</b> - If a Power Controller is implemented, this bit when written sets the power state of the slot per the defined encodings. Reads of this bit must reflect the value from the latest write,</p>	<u>RW</u>								

Bit Location	Register Description	Attributes
	<p>even if the corresponding hot-plug command is not complete, unless software issues a write, if required to, without waiting for the previous command to complete in which case the read value is undefined.</p> <p>Note that in some cases the power controller may autonomously remove slot power or not respond to a power-up request based on a detected fault condition, independent of the <u>Power Controller Control</u> setting.</p> <p>The defined encodings are:</p> <ul style="list-style-type: none"> <li><b>0b</b> Power On</li> <li><b>1b</b> Power Off</li> </ul> <p>If the <u>Power Controller Present</u> bit in the <u>Slot Capabilities Register</u> is Clear, then writes to this bit have no effect and the read value of this bit is undefined.</p>	
11	<p><b>Electromechanical Interlock Control</b> - If an Electromechanical Interlock is implemented, a write of 1b to this bit causes the state of the interlock to toggle. A write of 0b to this bit has no effect. A read of this bit always returns a 0b.</p>	RW
12	<p><b>Data Link Layer State Changed Enable</b> - If the <u>Data Link Layer Link Active Reporting Capable</u> is 1b, this bit enables software notification when <u>Data Link Layer Link Active</u> bit is changed.</p> <p>If the <u>Data Link Layer Link Active Reporting Capable</u> bit 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>	RW
13	<p><b>Auto Slot Power Limit Disable</b> - When Set, this disables the automatic sending of a <u>Set_Slot_Power_Limit Message</u> when a Link transitions from a non-DL_Up status to a <u>DL_Up</u> status.</p> <p>Downstream ports that don't support DPC are permitted to hardwire this bit to 0.</p> <p>Default value of this bit is implementation specific.</p>	RW
14	<p><b>In-Band PD Disable</b> - When Set, this bit disables the in-band presence detect mechanism from affecting the <u>Presence Detect State</u> bit, allowing that bit to report out-of-band presence detect exclusively. Otherwise, the <u>Presence Detect State</u> bit reflects the logical OR of the in-band and out-of-band presence detect mechanisms.</p> <p>In addition, the In-Band PD Disable bit governs the Component Presence state for the <u>Downstream Component Presence</u> field in the <u>Link Status 2 Register</u>. See Section 7.5.3.20.</p> <p>This bit must be implemented if the <u>In-Band PD Disable Supported</u> bit is 1b. Otherwise, this bit must be hardwired to 0b.</p> <p>Default value of this bit is 0b.</p>	RW

### 7.5.3.11 Slot Status Register (Offset 1Ah)

The Slot Status Register provides information about PCI Express Slot specific parameters. Figure 7-32 details allocation of register fields in the Slot Status Register; Table 7-27 provides the respective bit definitions. Register fields for status bits not implemented by the device have the RsvdZ attribute.

If this register is implemented but the Slot Implemented bit is Clear, the field behavior of this entire register with the exception of the Data Link Layer State Changed bit is undefined.

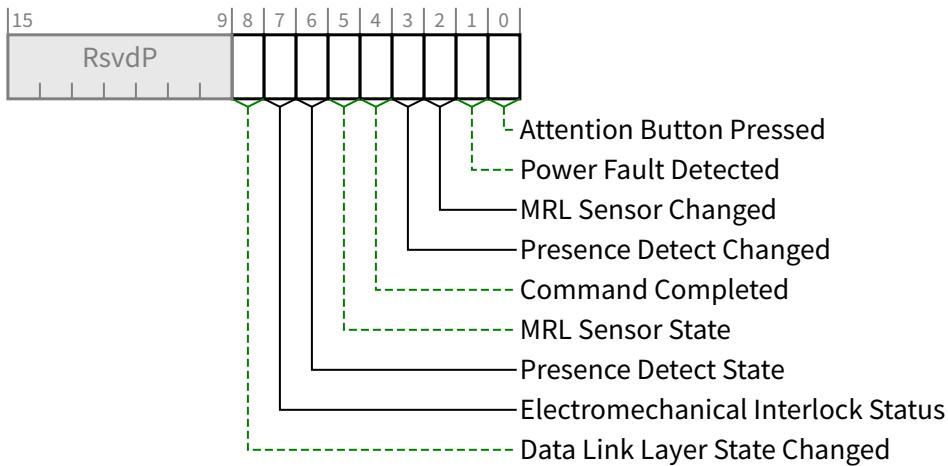


Figure 7-32 Slot Status Register

Table 7-27 Slot Status Register

Bit Location	Register Description	Attributes
0	<b>Attention Button Pressed</b> - If an Attention Button is implemented, this bit is Set when the attention button is pressed. If an Attention Button is not supported, this bit must not be Set.	<u>RW1C</u>
1	<b>Power Fault Detected</b> - If a Power Controller that supports power fault detection is implemented, this bit is Set when the Power Controller detects a power fault at this slot. Note that, depending on hardware capability, it is possible that a power fault can be detected at any time, independent of the <u>Power Controller Control</u> setting or the occupancy of the slot. If power fault detection is not supported, this bit must not be Set.	<u>RW1C</u>
2	<b>MRL Sensor Changed</b> - If an MRL sensor is implemented, this bit is Set when a <u>MRL Sensor State</u> change is detected. If an MRL sensor is not implemented, this bit must not be Set.	<u>RW1C</u>
3	<b>Presence Detect Changed</b> - This bit is Set when the value reported in the <u>Presence Detect State</u> bit is changed.	<u>RW1C</u>
4	<b>Command Completed</b> - If Command Completed notification is supported (if the <u>No Command Completed Support</u> bit in the Slot Capabilities Register is 0b), this bit is Set when a hot-plug command has completed and the Hot-Plug Controller is ready to accept a subsequent command. The <u>Command Completed</u> status bit is Set as an indication to host software that the Hot-Plug Controller has processed the previous command and is ready to receive the next command; it provides no guarantee that the action corresponding to the command is complete.  If Command Completed notification is not supported, this bit must be hardwired to 0b.	<u>RW1C</u>
5	<b>MRL Sensor State</b> - This bit reports the status of the MRL sensor if implemented.  Defined encodings are:  <b>0b</b> MRL Closed <b>1b</b> MRL Open	<u>RO</u>
6	<b>Presence Detect State</b> - This bit indicates the presence of an adapter in the slot. When the <u>In-Band PD Disable</u> bit is Clear, this is reflected by the logical “OR” of the Physical Layer in-band presence detect	<u>RO</u>

Bit Location	Register Description	Attributes
	<p>mechanism and, if present, any out-of-band presence detect mechanism defined for the slot's corresponding form factor. Note that the in-band presence detect mechanism requires that power be applied to an adapter for its presence to be detected. Consequently, form factors that require a power controller for hot-plug must implement an out-of-band presence detect mechanism. When the <u>In-Band PD Disable</u> bit is Set, the in-band presence detect mechanism has no effect on this bit.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>0b</b> Adapter not Present</li> <li><b>1b</b> Adapter Present</li> </ul> <p>This bit must be implemented on all Downstream Ports that implement slots. For Downstream Ports not connected to slots (where the <u>Slot Implemented</u> bit of the <u>PCI Express Capabilities Register</u> is 0b), this bit must be hardwired to 1b.</p>	
7	<p><b>Electromechanical Interlock Status</b> - If an Electromechanical Interlock is implemented, this bit indicates the status of the Electromechanical Interlock.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>0b</b> Electromechanical Interlock Disengaged</li> <li><b>1b</b> Electromechanical Interlock Engaged</li> </ul>	RO
8	<p><b>Data Link Layer State Changed</b> - This bit is Set when the value reported in the <u>Data Link Layer Link Active</u> bit of the <u>Link Status Register</u> is changed.</p> <p>In response to a <u>Data Link Layer State Changed</u> event, software must read the <u>Data Link Layer Link Active</u> bit of the <u>Link Status Register</u> to determine if the Link is active before initiating configuration cycles to the hot plugged device.</p>	RW1C

## IMPLEMENTATION NOTE

### No Slot Power Controller

For slots that do not implement a power controller, software must ensure that system power planes are enabled to provide power to slots prior to reading Presence Detect State.

#### 7.5.3.12 Root Control Register (Offset 1Ch)

The Root Control Register controls PCI Express Root Complex specific parameters. Figure 7-33 details allocation of register fields in the Root Control Register; Table 7-28 provides the respective bit definitions.

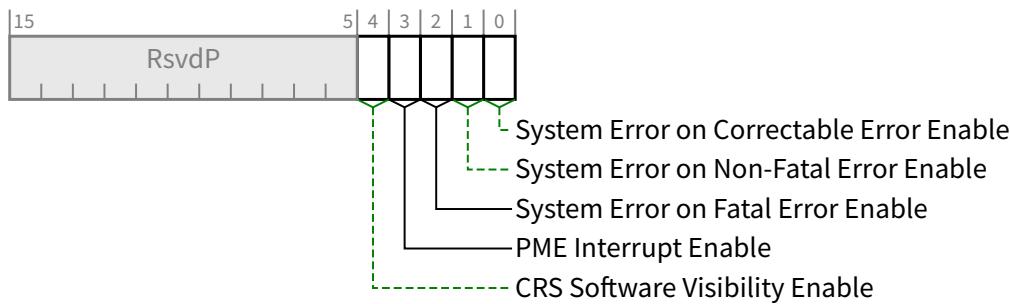


Figure 7-33 Root Control Register

Table 7-28 Root Control Register

Bit Location	Register Description	Attributes
0	<p><b>System Error on Correctable Error Enable</b> - If Set, this bit indicates that a System Error should be generated if a correctable error (ERR_COR) is reported by any of the devices in the Hierarchy Domain associated with this Root Port, or by the Root Port itself. The mechanism for signaling a System Error to the system is system specific.</p> <p>Root Complex Event Collectors provide support for the above-described functionality for RCiEPs.</p> <p>Default value of this bit is 0b.</p>	RW
1	<p><b>System Error on Non-Fatal Error Enable</b> - If Set, this bit indicates that a System Error should be generated if a Non-fatal error (ERR_NONFATAL) is reported by any of the devices in the Hierarchy Domain associated with this Root Port, or by the Root Port itself. The mechanism for signaling a System Error to the system is system specific.</p> <p>Root Complex Event Collectors provide support for the above-described functionality for RCiEPs.</p> <p>Default value of this bit is 0b.</p>	RW
2	<p><b>System Error on Fatal Error Enable</b> - If Set, this bit indicates that a System Error should be generated if a Fatal error (ERR_FATAL) is reported by any of the devices in the Hierarchy Domain associated with this Root Port, or by the Root Port itself. The mechanism for signaling a System Error to the system is system specific.</p> <p>Root Complex Event Collectors provide support for the above-described functionality for RCiEPs.</p> <p>Default value of this bit is 0b.</p>	RW
3	<p><b>PME Interrupt Enable</b> - When Set, this bit enables PME interrupt generation upon receipt of a PME Message as reflected in the PME Status bit (see Table 7-30). A PME interrupt is also generated if the PME Status bit is Set when this bit is changed from Clear to Set (see Section 5.3.3).</p> <p>Default value of this bit is 0b.</p>	RW
4	<p><b>CRS Software Visibility Enable</b> - When Set, this bit enables the Root Port to return Configuration Request Retry Status (CRS) Completion Status to software (see Section 2.3.1).</p> <p>Root Ports that do not implement this capability must hardwire this bit to 0b.</p> <p>Default value of this bit is 0b.</p>	RW

### 7.5.3.13 Root Capabilities Register (Offset 1Eh)

The Root Capabilities Register identifies PCI Express Root Port specific capabilities. Figure 7-34 details allocation of register fields in the Root Capabilities Register; Table 7-29 provides the respective bit definitions.

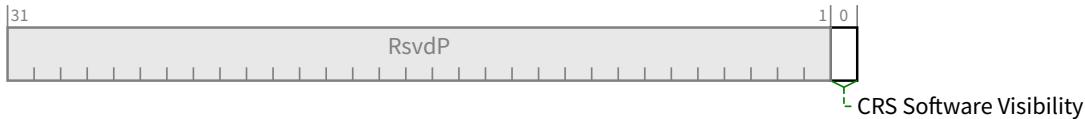


Figure 7-34 Root Capabilities Register

Table 7-29 Root Capabilities Register

Bit Location	Register Description	Attributes
0	<b>CRS Software Visibility</b> - When Set, this bit indicates that the Root Port is capable of returning Configuration Request Retry Status (CRS) Completion Status to software (see Section 2.3.1).	RO

### 7.5.3.14 Root Status Register (Offset 20h)

The Root Status Register provides information about PCI Express device specific parameters. Figure 7-35 details allocation of register fields in the Root Status Register; Table 7-30 provides the respective bit definitions.

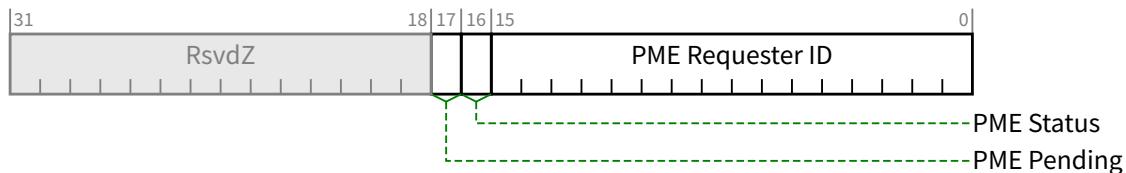


Figure 7-35 Root Status Register

Table 7-30 Root Status Register

Bit Location	Register Description	Attributes
15:0	<b>PME Requester ID</b> - This field indicates the PCI Requester ID of the last PME Requester. This field is only valid when the <u>PME Status</u> bit is Set.	RO
16	<b>PME Status</b> - This bit indicates that PME was asserted by the PME Requester indicated in the <u>PME Requester ID</u> field. Subsequent PMEs are kept pending until the status register is cleared by software by writing a 1b. Default value of this bit is 0b.	RW1C
17	<b>PME Pending</b> - This bit indicates that another PME is pending when the <u>PME Status</u> bit is Set. When the <u>PME Status</u> bit is cleared by software; the PME is delivered by hardware by setting the <u>PME Status</u> bit	RO

Bit Location	Register Description	Attributes
	again and updating the <u>PME Requester ID</u> field appropriately. The <u>PME Pending</u> bit is cleared by hardware if no more PMEs are pending.	

### 7.5.3.15 Device Capabilities 2 Register (Offset 24h)

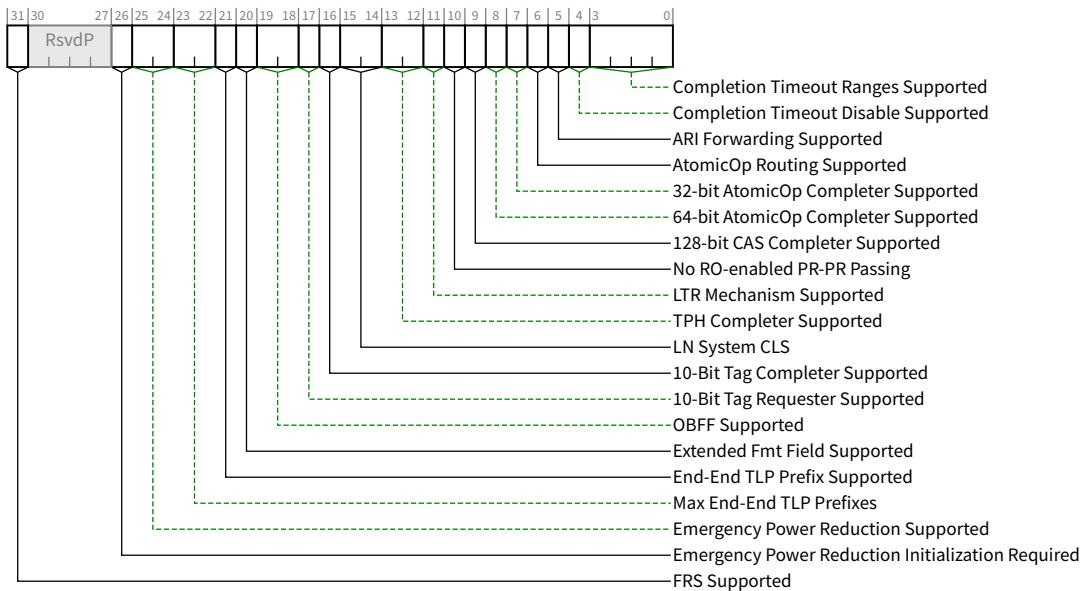


Figure 7-36 Device Capabilities 2 Register

Table 7-31 Device Capabilities 2 Register

Bit Location	Register Description	Attributes												
3:0	<p><b>Completion Timeout Ranges Supported</b> - This field indicates device Function support for the optional Completion Timeout programmability mechanism. This mechanism allows system software to modify the <u>Completion Timeout Value</u>.</p> <p>This field is applicable only to Root Ports, Endpoints that issue Requests on their own behalf, and PCI Express to PCI/PCI-X Bridges that take ownership of Requests issued on PCI Express. For all other Functions this field is Reserved and must be hardwired to 0000b.</p> <p>Four time value ranges are defined:</p> <table> <tr> <td><b>Range A</b></td> <td>50 µs to 10 ms</td> </tr> <tr> <td><b>Range B</b></td> <td>10 ms to 250 ms</td> </tr> <tr> <td><b>Range C</b></td> <td>250 ms to 4 s</td> </tr> <tr> <td><b>Range D</b></td> <td>4 s to 64 s</td> </tr> </table> <p>Bits are set according to the table below to show timeout value ranges supported.</p> <table> <tr> <td><b>0000b</b></td> <td>Completion Timeout programming not supported - the Function must implement a timeout value in the range 50 µs to 50 ms.</td> </tr> <tr> <td><b>0001b</b></td> <td>Range A</td> </tr> </table>	<b>Range A</b>	50 µs to 10 ms	<b>Range B</b>	10 ms to 250 ms	<b>Range C</b>	250 ms to 4 s	<b>Range D</b>	4 s to 64 s	<b>0000b</b>	Completion Timeout programming not supported - the Function must implement a timeout value in the range 50 µs to 50 ms.	<b>0001b</b>	Range A	HwInit
<b>Range A</b>	50 µs to 10 ms													
<b>Range B</b>	10 ms to 250 ms													
<b>Range C</b>	250 ms to 4 s													
<b>Range D</b>	4 s to 64 s													
<b>0000b</b>	Completion Timeout programming not supported - the Function must implement a timeout value in the range 50 µs to 50 ms.													
<b>0001b</b>	Range A													

Bit Location	Register Description	Attributes
	<p><b>0010b</b> Range B  <b>0011b</b> Ranges A and B  <b>0110b</b> Ranges B and C  <b>0111b</b> Ranges A, B, and C  <b>1110b</b> Ranges B, C, and D  <b>1111b</b> Ranges A, B, C, and D</p> <p>All other values are Reserved.</p> <p>It is strongly recommended that the Completion Timeout mechanism not expire in less than 10 ms.</p>	
4	<p><b>Completion Timeout Disable Supported</b> - A value of 1b indicates support for the Completion Timeout Disable mechanism.</p> <p>The Completion Timeout Disable mechanism is required for Endpoints that issue Requests on their own behalf and PCI Express to PCI/PCI-X Bridges that take ownership of Requests issued on PCI Express.</p> <p>This mechanism is optional for Root Ports.</p> <p>For all other Functions this field is Reserved and must be hardwired to 0b.</p>	<u>RO</u>
5	<p><b>ARI Forwarding Supported</b> - Applicable only to Switch Downstream Ports and Root Ports; must be 0b for other Function types. This bit must be set to 1b if a Switch Downstream Port or Root Port supports this optional capability. See <a href="#">Section 6.13</a> for additional details.</p>	<u>RO</u>
6	<p><b>AtomicOp Routing Supported</b> - Applicable only to Switch Upstream Ports, Switch Downstream Ports, and Root Ports; must be 0b for other Function types. This bit must be set to 1b if the Port supports this optional capability. See <a href="#">Section 6.15</a> for additional details.</p>	<u>RO</u>
7	<p><b>32-bit AtomicOp Completer Supported</b> - Applicable to Functions with Memory Space BARs as well as all Root Ports; must be 0b otherwise. Includes FetchAdd, Swap, and CAS AtomicOps. This bit must be set to 1b if the Function supports this optional capability. See <a href="#">Section 6.15.3.1</a> for additional RC requirements.</p>	<u>RO</u>
8	<p><b>64-bit AtomicOp Completer Supported</b> - Applicable to Functions with Memory Space BARs as well as all Root Ports; must be 0b otherwise. Includes FetchAdd, Swap, and CAS AtomicOps. This bit must be set to 1b if the Function supports this optional capability. See <a href="#">Section 6.15.3.1</a> for additional RC requirements.</p>	<u>RO</u>
9	<p><b>128-bit CAS Completer Supported</b> - Applicable to Functions with Memory Space BARs as well as all Root Ports; must be 0b otherwise. This bit must be set to 1b if the Function supports this optional capability. See <a href="#">Section 6.15</a> for additional details.</p>	<u>RO</u>
10	<p><b>No RO-enabled PR-PR Passing</b> - If this bit is Set, the routing element never carries out the passing permitted by <a href="#">Table 2-40</a> entry A2b that is associated with the Relaxed Ordering Attribute field being Set.</p> <p>This bit applies only for Switches and RCs that support peer-to-peer traffic between Root Ports. This bit applies only to Posted Requests being forwarded through the Switch or RC and does not apply to traffic originating or terminating within the Switch or RC itself. All Ports on a Switch or RC must report the same value for this bit.</p> <p>For all other functions, this bit must be 0b.</p>	<u>HwInit</u>
11	<p><b>LTR Mechanism Supported</b> - A value of 1b indicates support for the optional Latency Tolerance Reporting (LTR) mechanism.</p> <p>Root Ports, Switches and Endpoints are permitted to implement this capability.</p> <p>For a <a href="#">Multi-Function Device</a> associated with an Upstream Port, each Function must report the same value for this bit.</p> <p>For Bridges and other Functions that do not implement this capability, this bit must be hardwired to 0b.</p>	<u>RO</u>

Bit Location	Register Description	Attributes
13:12	<p><b>TPH Completer Supported</b> - Value indicates Completer support for TPH or Extended TPH. Applicable only to Root Ports and Endpoints. For all other Functions, this field is Reserved.</p> <p>Defined Encodings are:</p> <ul style="list-style-type: none"> <li><b>00b</b> TPH and Extended TPH Completer not supported.</li> <li><b>01b</b> TPH Completer supported; Extended TPH Completer not supported.</li> <li><b>10b</b> Reserved.</li> <li><b>11b</b> Both TPH and Extended TPH Completer supported.</li> </ul> <p>See Section 6.17 for details.</p>	RO
15:14	<p><b>LN System CLS</b> - Applicable only to Root Ports and RCRBs; must be 00b for all other Function types. This field indicates if the Root Port or RCRB supports LN protocol as an LN Completer, and if so, what cache line size is in effect.</p> <p>Encodings are:</p> <ul style="list-style-type: none"> <li><b>00b</b> LN Completer either not supported or not in effect</li> <li><b>01b</b> LN Completer with 64-byte cachelines in effect</li> <li><b>10b</b> LN Completer with 128-byte cachelines in effect</li> <li><b>11b</b> Reserved</li> </ul>	HwInit
16	<p><b>10-Bit Tag Completer Supported</b> - If this bit is Set, the Function supports 10-Bit Tag Completer capability; otherwise, the Function does not. See Section 2.2.6.2 .</p>	HwInit
17	<p><b>10-Bit Tag Requester Supported</b> - If this bit is Set, the Function supports 10-Bit Tag Requester capability; otherwise, the Function does not.</p> <p>This bit must not be Set if the <u>10-Bit Tag Completer Supported</u> bit is Clear.</p> <p>If the Function is an RCiEP, this bit must be Clear if the RC does not support 10-Bit Tag Completer capability for Requests coming from this RCiEP.</p> <p>Note that 10-Bit Tag field generation must be enabled by the <u>10-Bit Tag Requester Enable</u> bit in the Device Control 2 Register of the Requester Function before 10-Bit Tags can be generated by the Requester. See Section 2.2.6.2 .</p>	HwInit
19:18	<p><b>OBFF Supported</b> - This field indicates if OBFF is supported and, if so, what signaling mechanism is used.</p> <ul style="list-style-type: none"> <li><b>00b</b> OBFF Not Supported</li> <li><b>01b</b> OBFF supported using Message signaling only</li> <li><b>10b</b> OBFF supported using WAKE# signaling only</li> <li><b>11b</b> OBFF supported using WAKE# and Message signaling</li> </ul> <p>The value reported in this field must indicate support for WAKE# signaling only if:</p> <ul style="list-style-type: none"> <li>• for a Downstream Port, driving the WAKE# signal for OBFF is supported and the connector or component connected Downstream is known to receive that same WAKE# signal</li> <li>• for an Upstream Port, receiving the WAKE# signal for OBFF is supported and, if the component is on an add-in-card, that the component is connected to the WAKE# signal on the connector.</li> </ul> <p>Root Ports, Switch Ports, and Endpoints are permitted to implement this capability.</p> <p>For a Multi-Function Device associated with an Upstream Port, each Function must report the same value for this field.</p> <p>For Bridges and Ports that do not implement this capability, this field must be hardwired to 00b.</p>	HwInit

Bit Location	Register Description	Attributes
20	<p><b>Extended Fmt Field Supported</b> - If Set, the Function supports the 3-bit definition of the Fmt field. If Clear, the Function supports a 2-bit definition of the Fmt field. See <a href="#">Section 2.2</a>.</p> <p>Must be Set for Functions that support End-End TLP Prefixes. All Functions in an Upstream Port must have the same value for this bit. Each Downstream Port of a component may have a different value for this bit.</p> <p>It is strongly recommended that Functions support the 3-bit definition of the Fmt field.</p>	RO
21	<p><b>End-End TLP Prefix Supported</b> - Indicates whether End-End TLP Prefix support is offered by a Function. Values are:</p> <ul style="list-style-type: none"> <li><b>0b</b> No Support</li> <li><b>1b</b> Support is provided to receive TLPs containing End-End TLP Prefixes.</li> </ul> <p>All Ports of a Switch must have the same value for this bit.</p>	HwInit
23:22	<p><b>Max End-End TLP Prefixes</b> - Indicates the maximum number of End-End TLP Prefixes supported by this Function. See <a href="#">Section 2.2.10.2</a> for important details. Values are:</p> <ul style="list-style-type: none"> <li><b>01b</b> 1 End-End TLP Prefix</li> <li><b>10b</b> 2 End-End TLP Prefixes</li> <li><b>11b</b> 3 End-End TLP Prefixes</li> <li><b>00b</b> 4 End-End TLP Prefixes</li> </ul> <p>If <a href="#">End-End TLP Prefix Supported</a> is Clear, this field is <a href="#">RsvdP</a>.</p> <p>Different Root Ports that have the <a href="#">End-End TLP Prefix Supported</a> bit Set are permitted to report different values for this field.</p> <p>For Switches where <a href="#">End-End TLP Prefix Supported</a> is Set, this field must be 00b indicating support for up to four End-End TLP Prefixes.</p>	HwInit
25:24	<p><b>Emergency Power Reduction Supported</b> - Indicates support level of the optional <a href="#">Emergency Power Reduction State</a> feature. A Function can enter <a href="#">Emergency Power Reduction State</a> autonomously, or based on one of two mechanisms defined by the associated Form Factor Specification. Functions that are in the <a href="#">Emergency Power Reduction State</a> consume less power. The Emergency Power Reduction mechanism permits a chassis to request add-in cards to rapidly enter <a href="#">Emergency Power Reduction State</a> without involving system software. See <a href="#">Section 6.25</a> for additional details.</p> <p>Values are:</p> <ul style="list-style-type: none"> <li><b>00b</b> <a href="#">Emergency Power Reduction State</a> not supported</li> <li><b>01b</b> <a href="#">Emergency Power Reduction State</a> is supported and is triggered by Device Specific mechanism(s)</li> <li><b>10b</b> <a href="#">Emergency Power Reduction State</a> is supported and is triggered either by the mechanism defined in the corresponding Form Factor specification or by Device Specific mechanism(s)</li> <li><b>11b</b> Reserved</li> </ul> <p>This field is <a href="#">RsvdP</a> in Functions that are not associated with an Upstream Port.</p> <p>For Multi-Function Devices associated with an Upstream Port, all Functions that report a non-zero value for this field, must report the same non-zero value for this field.</p> <p>Default value is 00b.</p> <p>After reset, once this field returns a non-zero value, it must continue to return the same non-zero value, until the next reset.</p>	HwInit
26	<p><b>Emergency Power Reduction Initialization Required</b> - If Set, the Function requires complete or partial initialization upon exit from the <a href="#">Emergency Power Reduction State</a>. If Clear, the Function requires no</p>	HwInit

Bit Location	Register Description	Attributes
	<p>software intervention to return to normal operation upon exit from the <a href="#">Emergency Power Reduction State</a>. See <a href="#">Section 6.25</a> for additional details.</p> <p>For <a href="#">Multi-Function Devices</a> associated with an Upstream Port, all Functions must report the same value for this bit.</p> <p>This bit is <a href="#">RsvdP</a> in Functions that are not associated with an Upstream Port.</p> <p>Default value is 0b.</p> <p>After reset, when this field returns a non-zero value, it must continue to return the same non-zero value.</p>	
31	<p><b>FRS Supported</b> - When Set, indicates support for the optional Function Readiness Status (FRS) capability.</p> <p>Must be Set for all Functions that support generation or reception capabilities of <a href="#">FRS Messages</a>.</p> <p>Must not be Set by Switch Functions that do not generate <a href="#">FRS Messages</a> on their own behalf.</p>	<a href="#">HwInit</a>

## IMPLEMENTATION NOTE

### Use of the No RO-enabled PR-PR Passing Bit

The No RO-enabled PR-PR Passing bit allows platforms to utilize PCI Express switching elements on the path between a requester and completer for requesters that could benefit from a slightly less relaxed ordering model. An example is a device that cannot ensure that multiple overlapping posted writes to the same address are outstanding at the same time. The method by which such a device is enabled to utilize this mode is beyond the scope of this specification.

### 7.5.3.16 Device Control 2 Register (Offset 28h)

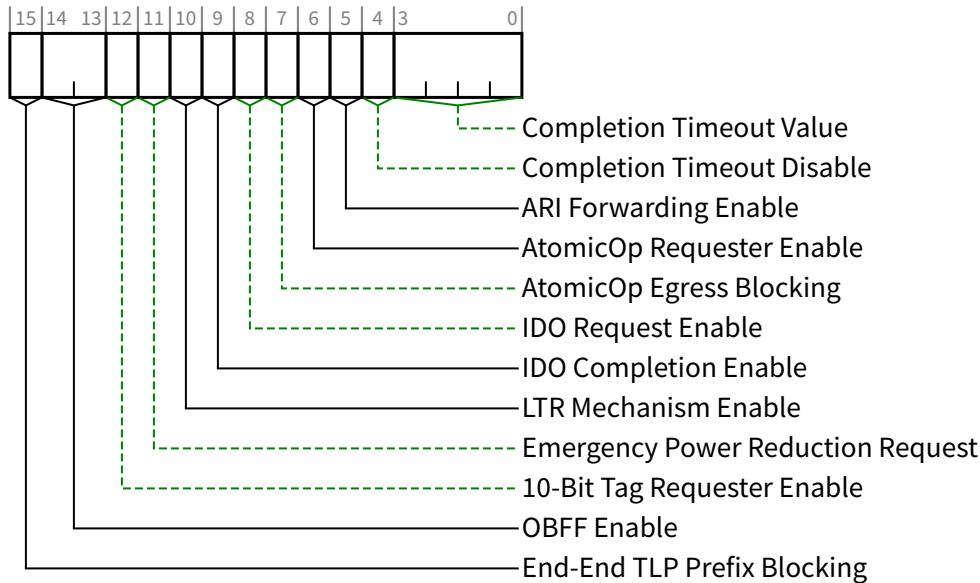


Figure 7-37 Device Control 2 Register

Table 7-32 Device Control 2 Register

Bit Location	Register Description	Attributes
3:0	<p><b>Completion Timeout Value</b> - In device Functions that support Completion Timeout programmability, this field allows system software to modify the <u>Completion Timeout Value</u>. This field is applicable to Root Ports, Endpoints that issue Requests on their own behalf, and PCI Express to PCI/PCI-X Bridges that take ownership of Requests issued on PCI Express. For all other Functions this field is Reserved and must be hardwired to 0000b.</p> <p>A Function that does not support this optional capability must hardwire this field to 0000b and is required to implement a timeout value in the range 50 µs to 50 ms. Functions that support Completion Timeout programmability must support the values given below corresponding to the programmability ranges indicated in the <u>Completion Timeout Ranges Supported</u> field.</p> <p>Defined encodings:</p> <ul style="list-style-type: none"> <li><b>0000b</b> Default range: 50 µs to 50 ms</li> </ul> <p>It is strongly recommended that the Completion Timeout mechanism not expire in less than 10 ms.</p> <p>Values available if Range A (50 µs to 10 ms) programmability range is supported:</p> <ul style="list-style-type: none"> <li><b>0001b</b> 50 µs to 100 µs</li> <li><b>0010b</b> 1 ms to 10 ms</li> </ul> <p>Values available if Range B (10 ms to 250 ms) programmability range is supported:</p> <ul style="list-style-type: none"> <li><b>0101b</b> 16 ms to 55 ms</li> <li><b>0110b</b> 65 ms to 210 ms</li> </ul> <p>Values available if Range C (250 ms to 4 s) programmability range is supported:</p>	RW

Bit Location	Register Description	Attributes
	<p><b>1001b</b> 260 ms to 900 ms  <b>1010b</b> 1 s to 3.5 s</p> <p>Values available if the Range D (4 s to 64 s) programmability range is supported:</p> <p><b>1101b</b> 4 s to 13 s  <b>1110b</b> 17 s to 64 s</p> <p>Values not defined above are Reserved.</p> <p>Software is permitted to change the value in this field at any time. For Requests already pending when the Completion Timeout Value is changed, hardware is permitted to use either the new or the old value for the outstanding Requests, and is permitted to base the start time for each Request either on when this value was changed or on when each request was issued.</p> <p>The default value for this field is 0000b.</p>	
4	<p><b>Completion Timeout Disable</b> - When Set, this bit disables the Completion Timeout mechanism.</p> <p>This bit is required for all Functions that support the Completion Timeout Disable Capability. Functions that do not support this optional capability are permitted to hardwire this bit to 0b</p> <p>Software is permitted to Set or Clear this bit at any time. When Set, the Completion Timeout detection mechanism is disabled. If there are outstanding Requests when the bit is cleared, it is permitted but not required for hardware to apply the completion timeout mechanism to the outstanding Requests. If this is done, it is permitted to base the start time for each Request on either the time this bit was cleared or the time each Request was issued.</p> <p>The default value for this bit is 0b.</p>	RW
5	<p><b>ARI Forwarding Enable</b> - When set, the Downstream Port disables its traditional Device Number field being 0 enforcement when turning a Type 1 Configuration Request into a Type 0 Configuration Request, permitting access to Extended Functions in an ARI Device immediately below the Port. See Section 6.13 .</p> <p>Default value of this bit is 0b. Must be hardwired to 0b if the ARI Forwarding Supported bit is 0b.</p> <p>This bit is not applicable and Reserved for Upstream Ports.</p>	RW / RsvdP
6	<p><b>AtomicOp Requester Enable</b> - Applicable only to Endpoints and Root Ports; must be hardwired to 0b for other Function types. The Function is allowed to initiate AtomicOp Requests only if this bit and the Bus Master Enable bit in the Command register are both Set.</p> <p>This bit is required to be RW if the Endpoint or Root Port is capable of initiating AtomicOp Requests, but otherwise is permitted to be hardwired to 0b.</p> <p>This bit does not serve as a capability bit. This bit is permitted to be RW even if no AtomicOp Requester capabilities are supported by the Endpoint or Root Port.</p> <p>Default value of this bit is 0b.</p>	RW
7	<p><b>AtomicOp Egress Blocking</b> - Applicable and mandatory for Switch Upstream Ports, Switch Downstream Ports, and Root Ports that implement AtomicOp routing capability; otherwise must be hardwired to 0b.</p> <p>When this bit is Set, AtomicOp Requests that target going out this Egress Port must be blocked. See Section 6.15.2 .</p> <p>Default value of this bit is 0b.</p>	RW
8	<p><b>IDO Request Enable</b> - If this bit is Set, the Function is permitted to set the ID-Based Ordering (IDO) bit (Attr[2]) of Requests it initiates (see Section 2.2.6.3 and Section 2.4 ).</p> <p>Endpoints, including RC Integrated Endpoints, and Root Ports are permitted to implement this capability.</p>	RW

Bit Location	Register Description	Attributes								
	<p>A Function is permitted to hardwire this bit to 0b if it never sets the IDO attribute in Requests. Default value of this bit is 0b.</p>									
9	<p><b>IDO Completion Enable</b> - If this bit is Set, the Function is permitted to set the <u>ID-Based Ordering (IDO)</u> bit (<u>Attr[2]</u>) of Completions it returns (see <u>Section 2.2.6.3</u> and <u>Section 2.4</u>). Endpoints, including RC Integrated Endpoints, and Root Ports are permitted to implement this capability.</p> <p>A Function is permitted to hardwire this bit to 0b if it never sets the IDO attribute in Completions. Default value of this bit is 0b.</p>	<u>RW</u>								
10	<p><b>LTR Mechanism Enable</b> - When Set to 1b, this bit enables Upstream Ports to send LTR messages and Downstream Ports to process LTR Messages.</p> <p>For a Multi-Function Device associated with an Upstream Port of a device that implements LTR, the bit in Function 0 is <u>RW</u>, and only Function 0 controls the component's Link behavior. In all other Functions of that device, this bit is <u>RsvdP</u>.</p> <p>Functions that do not implement the LTR mechanism are permitted to hardwire this bit to 0b.</p> <p>Default value of this bit is 0b.</p> <p>For Downstream Ports, this bit must be reset to the default value if the Port goes to <u>DL_Down</u> status.</p>	<u>RW/RsvdP</u>								
11	<p><b>Emergency Power Reduction Request</b> - If Set, all Functions in the component that support <u>Emergency Power Reduction State</u> must enter the <u>Emergency Power Reduction State</u>. If Clear these Functions must exit the <u>Emergency Power Reduction State</u> if no other reasons exist to preclude exiting this state. See <u>Section 6.25</u> for additional details.</p> <p>This bit is implemented in the lowest numbered Function associated with an Upstream Port that has a non-zero value in the <u>Emergency Power Reduction Supported</u> field. This bit is <u>RsvdP</u> in all other Functions.</p> <p>Default is 0b.</p>	<u>RW/RsvdP</u>								
12	<p><b>10-Bit Tag Requester Enable</b> - This bit, in combination with the <u>Extended Tag Field Enable</u> bit in the <u>Device Control Register</u>, determines how many Tag field bits a Requester is permitted to use. When the <u>10-Bit Tag Requester Enable</u> bit is Set, the Requester is permitted to use 10-Bit Tags. See <u>Section 2.2.6.2</u> for complete details.</p> <p>If software changes the value of this bit while the Function has outstanding Non-Posted Requests, the result is undefined.</p> <p>Functions that do not implement 10-Bit Tag Requester capability are permitted to hardwire this bit to 0b.</p> <p>Default value of this bit is 0b.</p>	<u>RW</u>								
14:13	<p><b>OBFF Enable</b> - This field enables the OBFF mechanism and selects the signaling method.</p> <table border="0"> <tr> <td><b>00b</b></td> <td>Disabled</td> </tr> <tr> <td><b>01b</b></td> <td>Enabled using Message signaling [Variation A]</td> </tr> <tr> <td><b>10b</b></td> <td>Enabled using Message signaling [Variation B]</td> </tr> <tr> <td><b>11b</b></td> <td>Enabled using WAKE# signaling</td> </tr> </table> <p>See <u>Section 6.19</u> for an explanation of the above encodings.</p> <p>This field is required for all Ports that support the OBFF Capability.</p> <p>For a Multi-Function Device associated with an Upstream Port of a Device that implements OBFF, the field in Function 0 is of type <u>RW</u>, and only Function 0 controls the Component's behavior. In all other Functions of that Device, this field is of type <u>RsvdP</u>.</p>	<b>00b</b>	Disabled	<b>01b</b>	Enabled using Message signaling [Variation A]	<b>10b</b>	Enabled using Message signaling [Variation B]	<b>11b</b>	Enabled using WAKE# signaling	<u>RW/RsvdP</u> (see description)
<b>00b</b>	Disabled									
<b>01b</b>	Enabled using Message signaling [Variation A]									
<b>10b</b>	Enabled using Message signaling [Variation B]									
<b>11b</b>	Enabled using WAKE# signaling									

Bit Location	Register Description	Attributes
	<p>Ports that do not implement OBFF are permitted to hardwire this field to 00b.</p> <p>Default value of this field is 00b.</p>	
15	<p><b>End-End TLP Prefix Blocking</b> - Controls whether the routing function is permitted to forward TLPs containing an End-End TLP Prefix. Values are:</p> <ul style="list-style-type: none"> <li><b>0b</b> Forwarding Enabled - Function is permitted to send TLPs with End-End TLP Prefixes.</li> <li><b>1b</b> Forwarding Blocked - Function is not permitted to send TLPs with End-End TLP Prefixes.</li> </ul> <p>This bit affects TLPs that exit the Switch/Root Complex using the associated Port. It does not affect TLPs forwarded internally within the Switch/Root Complex. It does not affect TLPs that enter through the associated Port, that originate in the associated Port or originate in a Root Complex Integrated Device integrated with the associated Port. As described in <a href="#">Section 2.2.10.2</a>, blocked TLPs are reported by the TLP Prefix Blocked Error.</p> <p>The default value of this bit is 0b.</p> <p>This bit is hardwired to 1b in Root Ports that support End-End TLP Prefixes but do not support forwarding of End-End TLP Prefixes.</p> <p>This bit is applicable to Root Ports and Switch Ports where the <a href="#">End-End TLP Prefix Supported</a> bit is Set. This bit is not applicable and is <a href="#">RsvdP</a> in all other cases.</p>	RW (see description)

### 7.5.3.17 Device Status 2 Register (Offset 2Ah)

This section is a placeholder. There are no capabilities that require this register.

This register must be treated by software as [RsvdZ](#).

### 7.5.3.18 Link Capabilities 2 Register (Offset 2Ch)

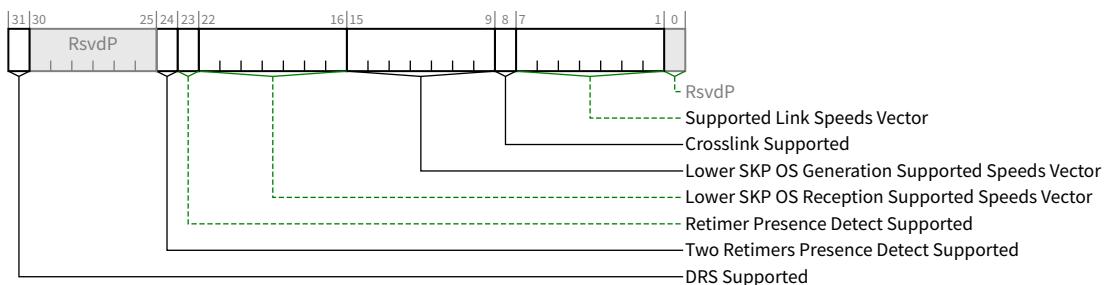


Figure 7-38 Link Capabilities 2 Register

Table 7-33 Link Capabilities 2 Register

Bit Location	Register Description	Attributes
7:1	<p><b>Supported Link Speeds Vector</b> - This field indicates the supported Link speed(s) of the associated Port. 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>	HwInit/RsvdP

Bit Location	Register Description	Attributes
	<p>Bit definitions within this field are:</p> <p><b>Bit 0</b> 2.5 GT/s  <b>Bit 1</b> 5.0 GT/s  <b>Bit 2</b> 8.0 GT/s  <b>Bit 3</b> 16.0 GT/s  <b>Bit 4</b> 32.0 GT/s  <b>Bits 6:5</b> RsvdP</p> <p><u>Multi-Function Devices</u> associated with an Upstream Port must report the same value in this field for all Functions.</p>	
8	<p><b>Crosslink Supported</b> - When set to 1b, this bit indicates that the associated Port supports crosslinks (see Section 4.2.6.3.1). When set to 0b on a Port that supports Link speeds of 8.0 GT/s or higher, this bit indicates that the associated Port does not support crosslinks. When set to 0b on a Port that only supports Link speeds of 2.5 GT/s or 5.0 GT/s, this bit provides no information regarding the Port's level of crosslink support.</p> <p>It is recommended that this bit be Set in any Port that supports crosslinks even though doing so is only required for Ports that also support operating at 8.0 GT/s or higher Link speeds.</p> <p>Note: Software should use this bit when referencing fields whose definition depends on whether or not the Port supports crosslinks (see Section 7.7.3.4).</p> <p><u>Multi-Function Devices</u> associated with an Upstream Port must report the same value in this field for all Functions.</p>	RO
15:9	<p><b>Lower SKP OS Generation Supported Speeds Vector</b> - If this field is non-zero, it indicates that the Port, when operating at the indicated speed(s) supports <u>SRIS</u> and also supports software control of the SKP Ordered Set transmission scheduling rate.</p> <p>Bit definitions within this field are:</p> <p><b>Bit 0</b> 2.5 GT/s  <b>Bit 1</b> 5.0 GT/s  <b>Bit 2</b> 8.0 GT/s  <b>Bit 3</b> 16.0 GT/s  <b>Bit 4</b> 32.0 GT/s  <b>Bits 6:5</b> RsvdP</p> <p><u>Multi-Function Devices</u> associated with an Upstream Port must report the same value in this field for all Functions.</p> <p>Behavior is undefined if a bit is Set in this field and the corresponding bit is not Set in the <u>Supported Link Speeds Vector</u>.</p>	HwInit/RsvdP
22:16	<p><b>Lower SKP OS Reception Supported Speeds Vector</b> - If this field is non-zero, it indicates that the Port, when operating at the indicated speed(s) supports <u>SRIS</u> and also supports receiving SKP OS at the rate defined for <u>SRNS</u> while running in <u>SRIS</u>.</p> <p>Bit definitions within this field are:</p> <p><b>Bit 0</b> 2.5 GT/s  <b>Bit 1</b> 5.0 GT/s  <b>Bit 2</b> 8.0 GT/s  <b>Bit 3</b> 16.0 GT/s</p>	HwInit/RsvdP

Bit Location	Register Description	Attributes
	<p><b>Bit 4</b> 32.0 GT/s</p> <p><b>Bits 6:5</b> RsvdP</p> <p>Multi-Function Devices associated with an Upstream Port must report the same value in this field for all Functions.</p> <p>Behavior is undefined if a bit is Set in this field and the corresponding bit is not Set in the <u>Supported Link Speeds Vector</u>.</p>	
23	<p><b>Retimer Presence Detect Supported</b> - When set to 1b, this bit indicates that the associated Port supports detection and reporting of Retimer presence.</p> <p>This bit must be set to 1b in a Port when the <u>Supported Link Speeds Vector</u> of the <u>Link Capabilities 2 Register</u> indicates support for a Link speed of 16.0 GT/s or higher.</p> <p>It is permitted to be set to 1b regardless of the supported Link speeds.</p> <p>Multi-Function Devices associated with an Upstream Port must report the same value in this field for all Functions.</p>	HwInit/RsvdP
24	<p><b>Two Retimers Presence Detect Supported</b> - When set to 1b, this bit indicates that the associated Port supports detection and reporting of two Retimers presence.</p> <p>This bit must be set to 1b in a Port when the <u>Supported Link Speeds Vector</u> of the <u>Link Capabilities 2 Register</u> indicates support for a Link speed of 16.0 GT/s or higher.</p> <p>It is permitted to be set to 1b regardless of the supported Link speeds if the <u>Retimer Presence Detect Supported</u> bit is also set to 1b.</p> <p>Multi-Function Devices associated with an Upstream Port must report the same value in this field for all Functions.</p>	HwInit/RsvdP
31	<p><b>DRS Supported</b> - When Set, indicates support for the optional Device Readiness Status (DRS) capability.</p> <p>Must be Set in Downstream Ports that support DRS.</p> <p>Must be Set in Downstream Ports that support FRS.</p> <p>For Upstream Ports that support DRS, it is strongly recommended that this bit be Set in Function 0. For all other Functions associated with an Upstream Port, this bit must be Clear.<sup>145</sup></p> <p>Must be Clear in Functions that are not associated with a Port.</p> <p>RsvdP in all other Functions.</p>	HwInit/RsvdP

<sup>145</sup>. It is expressly permitted for Upstream Ports to send DRS Messages even when the DRS Supported bit is Clear.

## IMPLEMENTATION NOTE

### Software Management of Link Speeds With Earlier Hardware

Hardware components compliant to versions prior to [PCIe-3.0] either did not implement the Link Capabilities 2 Register, or the register was Reserved.

For software to determine the supported Link speeds for components where the Link Capabilities 2 Register is either not implemented, or the value of its Supported Link Speeds Vector is 0000000b, software can read bits 3:0 of the 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 encoding of the values for the Current Link Speed field (in the Link Status Register) and Target Link Speed field (in the Link Control 2 Register) is the same as above.

## 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.

#### 7.5.3.19 Link Control 2 Register (Offset 30h)

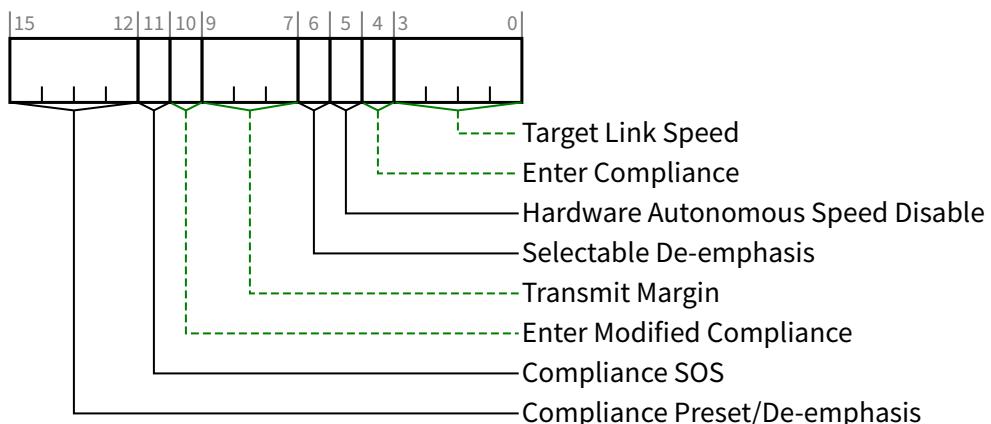


Figure 7-39 Link Control 2 Register

Table 7-34 Link Control 2 Register

Bit Location	Register Description	Attributes
3:0	<p><b>Target Link Speed</b> - For Downstream Ports, this field sets an upper limit on Link operational speed by restricting the values advertised by the Upstream component in its training sequences.</p> <p>The encoded value specifies a Bit Location in the <u>Supported Link Speeds Vector</u> (in the <u>Link Capabilities 2 Register</u>) that corresponds to the desired target 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>If a value is written to this field that does not correspond to a supported speed (as indicated by the <u>Supported Link Speeds Vector</u>), the result is undefined.</p> <p>If either of the <u>Enter Compliance</u> or <u>Enter Modified Compliance</u> bits are implemented, then this field must also be implemented.</p> <p>The default value of this field is the highest Link speed supported by the component (as reported in the <u>Max Link Speed</u> field of the <u>Link Capabilities Register</u>) unless the corresponding platform/form factor requires a different default value.</p> <p>For both Upstream and Downstream Ports, this field is used to set the target compliance mode speed when software is using the <u>Enter Compliance</u> bit to force a Link into compliance mode.</p> <p>For Upstream Ports, if the <u>Enter Compliance</u> bit is Clear, this field is permitted to have no effect.</p> <p>For a <u>Multi-Function Device</u> associated with an Upstream Port, the field in Function 0 is of type <u>RWS</u>, and only Function 0 controls the component's Link behavior. In all other Functions of that device, this field is of type <u>RsvdP</u>.</p> <p>Components that support only the 2.5 GT/s speed are permitted to hardwire this field to 0000b.</p>	<u>RWS/RsvdP</u> (see description)
4	<p><b>Enter Compliance</b> - Software is permitted to force a Link to enter Compliance mode (at the speed indicated in the Target Link Speed field and the de-emphasis/preset level indicated by the <u>Compliance Preset/De-emphasis</u> field) by setting this bit to 1b in both components on a Link and then initiating a hot reset on the Link.</p> <p>Default value of this bit following Fundamental Reset is 0b.</p> <p>For a <u>Multi-Function Device</u> associated with an Upstream Port, the bit in Function 0 is of type <u>RWS</u>, and only Function 0 controls the component's Link behavior. In all other Functions of that device, this bit is of type <u>RsvdP</u>.</p> <p>Components that support only the 2.5 GT/s speed are permitted to hardwire this bit to 0b.</p> <p>This bit is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this bit only during debug or compliance testing. In all other cases, the system must ensure that this bit is set to the default value.</p>	<u>RWS/RsvdP</u> (see description)
5	<p><b>Hardware Autonomous Speed Disable</b> - When Set, this bit disables hardware from changing the Link speed for device-specific reasons other than attempting to correct unreliable Link operation by reducing Link speed. Initial transition to the highest supported common link speed is not blocked by this bit.</p>	<u>RWS/RsvdP</u> (see description)

Bit Location	Register Description	Attributes				
	<p>For a <u>Multi-Function Device</u> associated with an Upstream Port, the bit in Function 0 is of type <u>RWS</u>, and only Function 0 controls the component's Link behavior. In all other Functions of that device, this bit is of type <u>RsvdP</u>.</p> <p>Functions that do not implement the associated mechanism are permitted to hardwire this bit to 0b.</p> <p>Default value of this bit is 0b.</p>					
6	<p><b>Selectable De-emphasis</b> - When the Link is operating at 5.0 GT/s speed, this bit is used to control the transmit de-emphasis of the link in specific situations. See <u>Section 4.2.6</u> for detailed usage information.</p> <p>Encodings:</p> <table> <tr> <td><b>1b</b></td> <td>-3.5 dB</td> </tr> <tr> <td><b>0b</b></td> <td>-6 dB</td> </tr> </table> <p>When the Link is not operating at 5.0 GT/s speed, the setting of this bit has no effect. Components that support only the 2.5 GT/s speed are permitted to hardwire this bit to 0b.</p> <p>This bit is not applicable and Reserved for Endpoints, PCI Express to PCI/PCI-X bridges, and Upstream Ports of Switches.</p>	<b>1b</b>	-3.5 dB	<b>0b</b>	-6 dB	HwInit
<b>1b</b>	-3.5 dB					
<b>0b</b>	-6 dB					
9:7	<p><b>Transmit Margin</b> - This field controls the value of the non-deemphasized voltage level at the Transmitter pins. This field is reset to 000b on entry to the <u>LTSSM Polling.Configuration</u> substate (see <u>Chapter 4</u> for details of how the Transmitter voltage level is determined in various states).</p> <p>Encodings:</p> <table> <tr> <td><b>000b</b></td> <td>Normal operating range</td> </tr> <tr> <td><b>001b-111b</b></td> <td>As defined in Section 8.3.4 not all encodings are required to be implemented.</td> </tr> </table> <p>For a <u>Multi-Function Device</u> associated with an Upstream Port, the field in Function 0 is of type <u>RWS</u>, and only Function 0 controls the component's Link behavior. In all other Functions of that device, this field is of type <u>RsvdP</u>.</p> <p>Default value of this field is 000b.</p> <p>Components that support only the 2.5 GT/s speed are permitted to hardwire this bit to 000b.</p> <p>This field is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this field only during debug or compliance testing. In all other cases, the system must ensure that this field is set to the default value.</p>	<b>000b</b>	Normal operating range	<b>001b-111b</b>	As defined in Section 8.3.4 not all encodings are required to be implemented.	RWS/RsvdP (see description)
<b>000b</b>	Normal operating range					
<b>001b-111b</b>	As defined in Section 8.3.4 not all encodings are required to be implemented.					
10	<p><b>Enter Modified Compliance</b> - When this bit is set to 1b, the device transmits Modified Compliance Pattern if the LTSSM enters <u>Polling.Compliance</u> substate.</p> <p>Components that support only the 2.5 GT/s speed are permitted to hardwire this bit to 0b.</p> <p>For a <u>Multi-Function Device</u> associated with an Upstream Port, the bit in Function 0 is of type <u>RWS</u>, and only Function 0 controls the component's Link behavior. In all other Functions of that device, this bit is of type <u>RsvdP</u>.</p> <p>Default value of this bit is 0b.</p> <p>This bit is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this bit only during debug or compliance testing. In all other cases, the system must ensure that this bit is set to the default value.</p>	RWS/RsvdP (see description)				
11	<p><b>Compliance SOS</b> - When set to 1b, the LTSSM is required to send SKP Ordered Sets between sequences when sending the Compliance Pattern or Modified Compliance Pattern.</p> <p>For a <u>Multi-Function Device</u> associated with an Upstream Port, the bit in Function 0 is of type <u>RWS</u>, and only Function 0 controls the component's Link behavior. In all other Functions of that device, this bit is of type <u>RsvdP</u>.</p>	RWS/RsvdP (see description)				

Bit Location	Register Description	Attributes				
	<p>The default value of this bit is 0b.</p> <p>This bit is applicable when the Link is operating at 2.5 GT/s or 5.0 GT/s data rates only.</p> <p>Components that support only the 2.5 GT/s speed are permitted to hardwire this bit to 0b.</p>					
15:12	<p><b>Compliance Preset/De-emphasis -</b></p> <p>For 8.0 GT/s and higher Data Rate: This field sets the Transmitter Preset in <u>Polling.Compliance</u> state if the entry occurred due to the <u>Enter Compliance</u> bit being 1b. The encodings are defined in <u>Section 4.2.3.2</u>. Results are undefined if a reserved preset encoding is used when entering <u>Polling.Compliance</u> in this way.</p> <p>For 5.0 GT/s Data Rate: This field sets the de-emphasis level in <u>Polling.Compliance</u> state if the entry occurred due to the <u>Enter Compliance</u> bit being 1b.</p> <p>Defined Encodings are:</p> <table> <tr> <td><b>0001b</b></td> <td>-3.5 dB</td> </tr> <tr> <td><b>0000b</b></td> <td>-6 dB</td> </tr> </table> <p>When the Link is operating at 2.5 GT/s, the setting of this field has no effect. Components that support only 2.5 GT/s speed are permitted to hardwire this field to 0000b.</p> <p>For a Multi-Function Device associated with an Upstream Port, the field in Function 0 is of type <u>RWS</u>, and only Function 0 controls the component's Link behavior. In all other Functions of that device, this field is of type <u>RsvdP</u>.</p> <p>The default value of this field is 0000b.</p> <p>This field is intended for debug and compliance testing purposes. System firmware and software is allowed to modify this field only during debug or compliance testing. In all other cases, the system must ensure that this field is set to the default value.</p>	<b>0001b</b>	-3.5 dB	<b>0000b</b>	-6 dB	RWS/RsvdP (see description)
<b>0001b</b>	-3.5 dB					
<b>0000b</b>	-6 dB					

## IMPLEMENTATION NOTE

### Selectable De-emphasis Usage

Selectable De-emphasis setting is applicable only to Root Ports and Switch Downstream Ports. The De-emphasis setting is implementation specific and depends on the platform or enclosure in which the Root Port or the Switch Downstream Port is located. System firmware or hardware strapping is used to configure the Selectable De-emphasis value. In cases where system firmware cannot be used to set the de-emphasis value (for example, a hot plugged Switch), hardware strapping must be used to set the de-emphasis value.

### 7.5.3.20 Link Status 2 Register (Offset 32h)

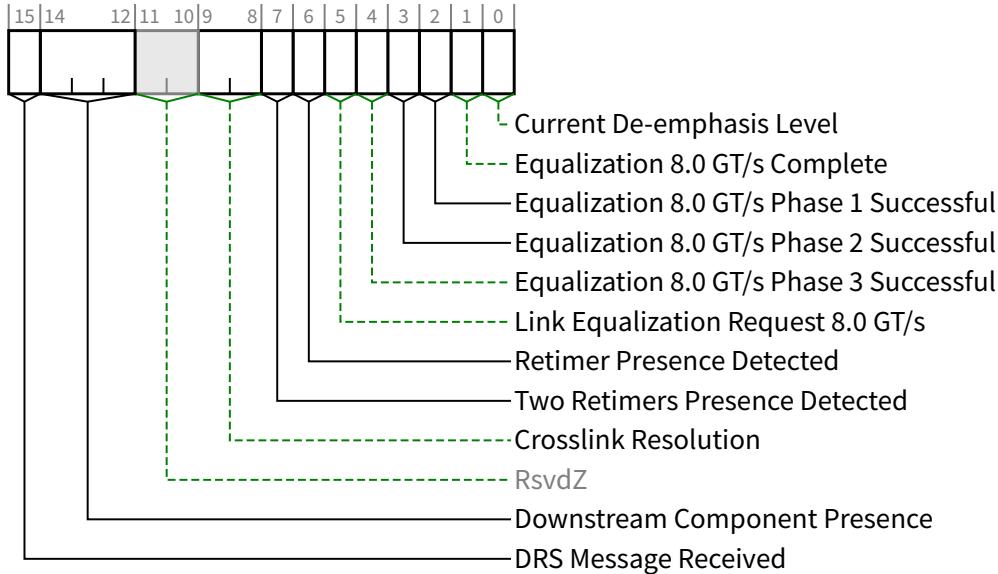


Figure 7-40 Link Status 2 Register

Table 7-35 Link Status 2 Register

Bit Location	Register Description	Attributes
0	<p><b>Current De-emphasis Level</b> - When the Link is operating at 5.0 GT/s speed, this bit reflects the level of de-emphasis.</p> <p>Encodings:</p> <ul style="list-style-type: none"> <li><b>1b</b> -3.5 dB</li> <li><b>0b</b> -6 dB</li> </ul> <p>The value in this bit is undefined when the Link is not operating at 5.0 GT/s speed.</p> <p>Components that support only the 2.5 GT/s speed are permitted to hardwire this bit to 0b.</p> <p>For components that support speeds greater than 2.5 GT/s, Multi-Function Devices associated with an Upstream Port must report the same value in this field for all Functions of the Port.</p>	RO
1	<p><b>Equalization 8.0 GT/s Complete</b> - When set to 1b, this bit indicates that the Transmitter Equalization procedure at the 8.0 GT/s data rate has completed. Details of the Transmitter Equalization process and when this bit needs to be set to 1b is provided in <a href="#">Section 4.2.6.4.2</a>.</p> <p>The default value of this bit is 0b.</p> <p>For Multi-Function Upstream Port, this bit must be implemented in Function 0 and RsvdZ in other Functions. Components that only support speeds below 8.0 GT/s are permitted to hardwire this bit to 0b.</p>	ROS
2	<p><b>Equalization 8.0 GT/s Phase 1 Successful</b> - When set to 1b, this bit indicates that Phase 1 of the 8.0 GT/s Transmitter Equalization procedure has successfully completed. Details of the Transmitter Equalization process and when this bit needs to be set to 1b is provided in <a href="#">Section 4.2.6.4.2</a>.</p>	ROS

Bit Location	Register Description	Attributes
	<p>The default value of this bit is 0b.</p> <p>For Multi-Function Upstream Port, this bit must be implemented in Function 0 and <u>RsvdZ</u> in other Functions. Components that only support speeds below 8.0 GT/s are permitted to hardwire this bit to 0b.</p>	
3	<p><b>Equalization 8.0 GT/s Phase 2 Successful</b> - When set to 1b, this bit indicates that Phase 2 of the 8.0 GT/s Transmitter Equalization procedure has successfully completed. Details of the Transmitter Equalization process and when this bit needs to be set to 1b is provided in <a href="#">Section 4.2.6.4.2</a>.</p> <p>The default value of this bit is 0b.</p> <p>For Multi-Function Upstream Port, this bit must be implemented in Function 0 and <u>RsvdZ</u> in other Functions. Components that only support speeds below 8.0 GT/s are permitted to hardwire this bit to 0b.</p>	ROS
4	<p><b>Equalization 8.0 GT/s Phase 3 Successful</b> - When set to 1b, this bit indicates that Phase 3 of the 8.0 GT/s Transmitter Equalization procedure has successfully completed. Details of the Transmitter Equalization process and when this bit needs to be set to 1b is provided in <a href="#">Section 4.2.6.4.2</a>.</p> <p>The default value of this bit is 0b.</p> <p>For Multi-Function Upstream Port, this bit must be implemented in Function 0 and <u>RsvdZ</u> in other Functions. Components that only support speeds below 8.0 GT/s are permitted to hardwire this bit to 0b.</p>	ROS
5	<p><b>Link Equalization Request 8.0 GT/s</b> - This bit is Set by hardware to request the 8.0 GT/s Link equalization process to be performed on the Link. Refer to <a href="#">Section 4.2.3</a> and <a href="#">Section 4.2.6.4.2</a> for details.</p> <p>The default value of this bit is 0b.</p> <p>For Multi-Function Upstream Port, this bit must be implemented in Function 0 and <u>RsvdZ</u> in other Functions. Components that only support speeds below 8.0 GT/s are permitted to hardwire this bit to 0b.</p>	RW1CS
6	<p><b>Retimer Presence Detected</b> - When set to 1b, this bit indicates that a Retimer was present during the most recent Link negotiation. Refer to <a href="#">Section 4.2.6.3.5.1</a> for details.</p> <p>The default value of this bit is 0b.</p> <p>This bit is required for Ports that have the <u>Retimer Presence Detect Supported</u> bit of the <u>Link Capabilities 2 Register</u> set to 1b.</p> <p>Ports that have the <u>Retimer Presence Detect Supported</u> bit set to 0b are permitted to hardwire this bit to 0b.</p> <p>For Multi-Function Devices associated with an Upstream Port, this bit must be implemented in Function 0 and <u>RsvdZ</u> in all other Functions.</p>	ROS/ <u>RsvdZ</u>
7	<p><b>Two Retimers Presence Detected</b> - When set to 1b, this bit indicates that two Retimers were present during the most recent Link negotiation. Refer to <a href="#">Section 4.2.6.3.5.1</a> for details.</p> <p>The default value of this bit is 0b.</p> <p>This bit is required for Ports that have the <u>Two Retimers Presence Detect Supported</u> bit of the <u>Link Capabilities 2 Register</u> set to 1b.</p> <p>Ports that have the <u>Two Retimers Presence Detect Supported</u> bit set to 0b are permitted to hardwire this bit to 0b.</p> <p>For Multi-Function Devices associated with an Upstream Port, this bit must be implemented in Function 0 and <u>RsvdZ</u> in all other Functions.</p>	ROS/ <u>RsvdZ</u>

Bit Location	Register Description	Attributes
9:8	<p><b>Crosslink Resolution</b> - This field indicates the state of the Crosslink negotiation. It must be implemented if <u>Crosslink Supported</u> is Set and the Port supports 16.0 GT/s or higher data rate. It is permitted to be implemented in all other Ports. If <u>Crosslink Supported</u> is Clear, this field may be hardwired to 01b or 10b.</p> <p>Encoding is:</p> <ul style="list-style-type: none"> <li><b>00b</b> <u>Crosslink Resolution</u> is not supported. No information is provided regarding the status of the Crosslink negotiation.</li> <li><b>01b</b> Crosslink negotiation resolved as an Upstream Port.</li> <li><b>10b</b> Crosslink negotiation resolved as a Downstream Port.</li> <li><b>11b</b> Crosslink negotiation is not completed.</li> </ul> <p>Once a value of 01b or 10b is returned in this field, that value must continue to be returned while the Link is Up.</p>	RO
14:12	<p><b>Downstream Component Presence</b> - This field indicates the presence and DRS status for the Downstream Component, if any, connected to the Link; defined values are:</p> <ul style="list-style-type: none"> <li><b>000b</b> <b>Link Down - Presence Not Determined</b></li> <li><b>001b</b> <b>Link Down - Component Not Present</b> indicates the Downstream Port (DP) has determined that a Downstream Component is not present</li> <li><b>010b</b> <b>Link Down - Component Present</b> indicates the DP has determined that a Downstream Component is present, but the Data Link Layer is not active</li> <li><b>011b</b> Reserved</li> <li><b>100b</b> <b>Link Up - Component Present</b> indicates the DP has determined that a Downstream Component is present, but no DRS Message has been received since the Data Link Layer became active</li> <li><b>101b</b> <b>Link Up - Component Present and DRS Received</b> indicates the DP has received a DRS Message since the Data Link Layer became active</li> <li><b>110b</b> Reserved</li> <li><b>111b</b> Reserved</li> </ul> <p>Downstream Component Presence state must be determined by the logical “OR” of the Physical Layer in-band presence detect mechanism and, if present, any out-of-band presence detect mechanism implemented for the Link. If no out-of-band presence detect mechanism is implemented, then <u>Downstream Component Presence</u> state must be determined solely by the Physical Layer in-band presence detect mechanism.</p> <p>If the <u>In-Band PD Disable</u> bit in the <u>Slot Control Register</u> is Set, the Physical Layer in-band presence detect mechanism must always indicate that no component is present.</p> <p>Component Presence, Link Up, and DRS Received states indicated by this field must reflect their maskable states, which are controlled by the <u>SFI PD State Mask</u>, <u>SFI DLL State Mask</u>, or <u>SFI DRS Mask</u> bits in the <u>SFI Control Register</u>. See <u>Section 7.9.23.3</u>.</p> <p>This field must be implemented in any Downstream Port where the <u>DRS Supported</u> bit is Set in the <u>Link Capabilities 2 Register</u>.</p> <p>This field is <u>RsvdZ</u> for all other Functions.</p> <p>Default value of this field is 000b.</p>	RO/RsvdZ
15	<p><b>DRS Message Received</b> - This bit must be Set whenever the Port receives a DRS Message.</p> <p>This bit must be Cleared in <u>DL_Down</u>.</p> <p>This bit must be implemented in any Downstream Port where the <u>DRS Supported</u> bit is Set in the <u>Link Capabilities 2 Register</u>.</p>	RW1C/RsvdZ

Bit Location	Register Description	Attributes
	This bit is <u>RsvdZ</u> for all other Functions. Default value of this bit is 0b.	

### 7.5.3.21 Slot Capabilities 2 Register (Offset 34h)

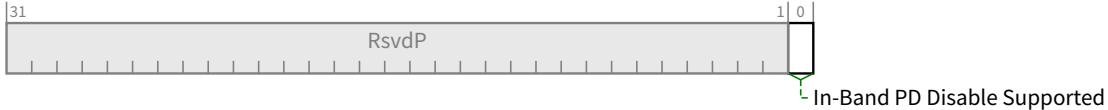


Figure 7-41 Slot Capabilities 2 Register

Table 7-36 Slot Capabilities 2 Register

Bit Location	Register Description	Attributes
0	<b>In-Band PD Disable Supported</b> - When Set, this bit indicates that this slot supports disabling the reporting of the in-band presence detect state, as controlled by the In-Band PD Disable bit in the Slot Control Register. If the slot does not support an out-of-band presence detect mechanism, this bit must be Clear.	HwInit

### 7.5.3.22 Slot Control 2 Register (Offset 38h)

This section is a placeholder. There are no capabilities that require this register.

This register must be treated by software as RsvdP.

### 7.5.3.23 Slot Status 2 Register (Offset 3Ah)

This section is a placeholder. There are no capabilities that require this register.

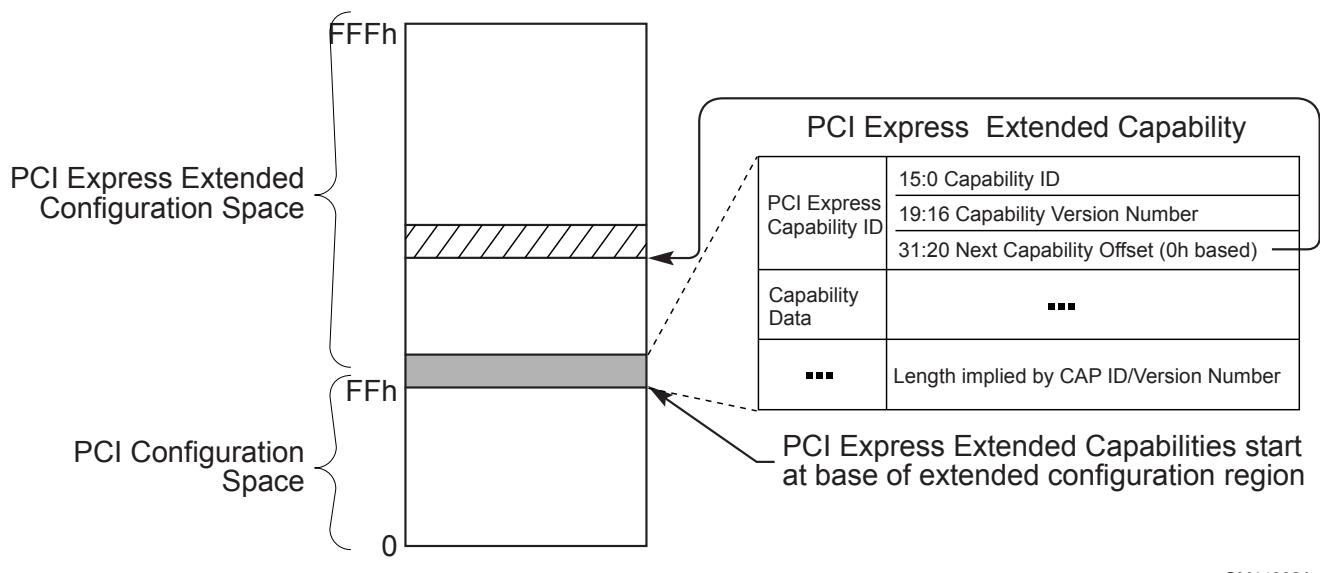
This register must be treated by software as RsvdZ.

## 7.6 PCI Express Extended Capabilities

PCI Express Extended Capability registers are located in Configuration Space at offsets 256 or greater as shown in Figure 7-42 or in the Root Complex Register Block (RCRB). These registers when located in the Configuration Space are accessible using only the PCI Express Enhanced Configuration Access Mechanism (ECAM).

PCI Express Extended Capability structures are allocated using a linked list of optional or required PCI Express Extended Capabilities following a format resembling PCI Capability structures. The first DWORD of the Capability structure identifies the Capability and version and points to the next Capability as shown in Figure 7-42 .

Each Capability structure must be DWORD aligned.



OM14302A

Figure 7-42 PCI Express Extended Configuration Space Layout

### 7.6.1 Extended Capabilities in Configuration Space

Extended Capabilities in Configuration Space always begin at offset 100h with a PCI Express Extended Capability header (Section 7.6.3). Absence of any Extended Capabilities is required to be indicated by an Extended Capability header with a Capability ID of 0000h, a Capability Version of 0h, and a Next Capability Offset of 000h.

### 7.6.2 Extended Capabilities in the Root Complex Register Block

Extended Capabilities in a Root Complex Register Block always begin at offset 000h with a PCI Express Extended Capability header (Section 7.6.3). Absence of any Extended Capabilities is required to be indicated by an Extended Capability header with a Capability ID of FFFFh and a Next Capability Offset of 000h.

### 7.6.3 PCI Express Extended Capability Header

All PCI Express Extended Capabilities must begin with a PCI Express Extended Capability Header. Figure 7-43 details the allocation of register fields of a PCI Express Extended Capability Header; Table 7-37 provides the respective bit definitions.

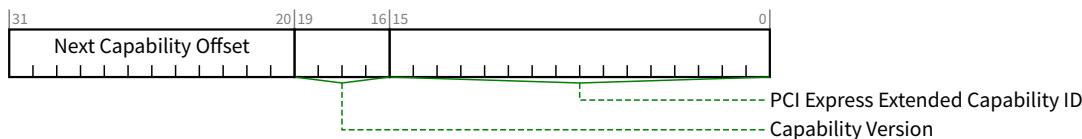


Figure 7-43 PCI Express Extended Capability Header

*Table 7-37 PCI Express 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.	<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.  A version of the specification that changes the Extended Capability in a way that is not otherwise identifiable (e.g., through a new Capability field) is permitted to increment this field. All such changes to the Capability structure must be software-compatible. Software must check for Capability Version numbers that are greater than or equal to the highest number defined when the software is written, as Functions reporting any such Capability Version numbers will contain a Capability structure that is compatible with that piece of software.	<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>

## 7.7 PCI and PCIe Capabilities Required by the Base Spec in Some Situations

The following capabilities are required by this specification for some Functions. For example, Functions that support specific data rates, functions that generate interrupts, etc.

### 7.7.1 MSI Capability Structures

All PCI Express device Functions that are capable of generating interrupts must implement MSI or MSI-X or both.

The MSI Capability structure is described in this section. The MSI-X Capability structure is described in [Section 7.7.2](#).

The MSI Capability structure is illustrated in [Figure 7-44](#) and [Figure 7-45](#). Each device Function that supports MSI (in a Multi-Function Device) must implement its own MSI Capability structure. More than one MSI Capability structure per Function is prohibited, but a Function is permitted to have both an MSI and an MSI-X Capability structure.

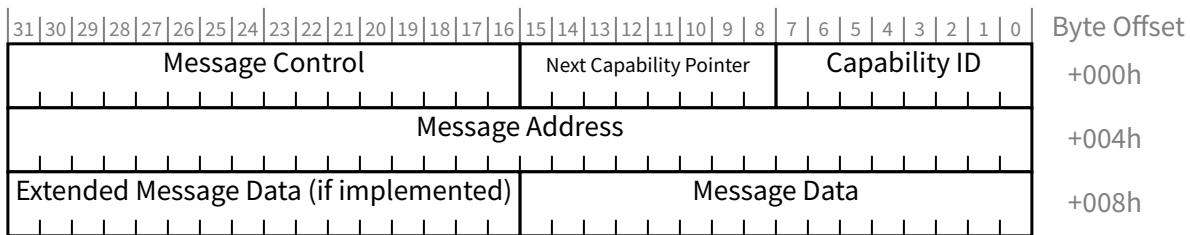


Figure 7-44 MSI Capability Structure for 32-bit Message Address

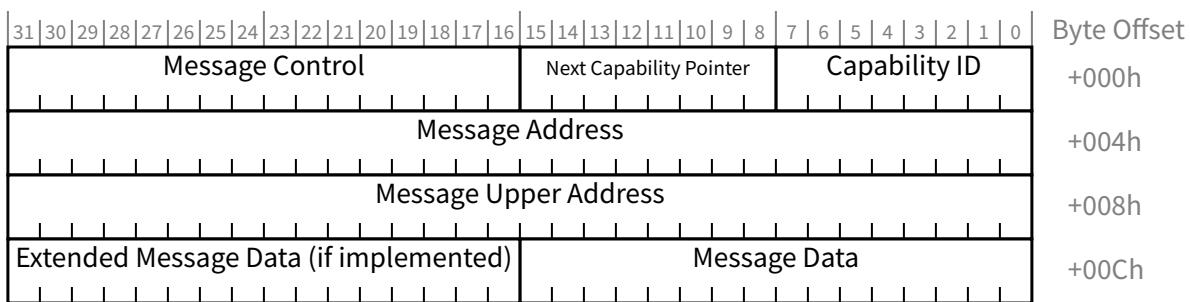


Figure 7-45 MSI Capability Structure for 64-bit Message Address

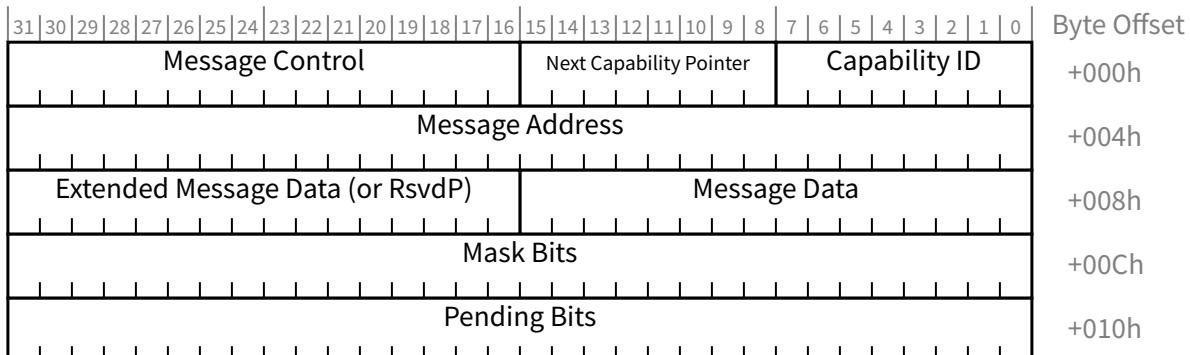
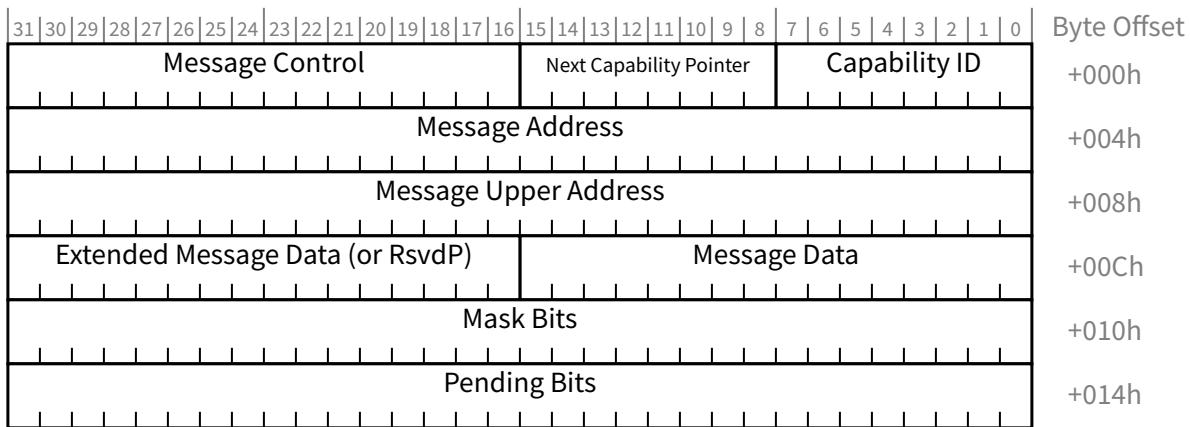


Figure 7-46 MSI Capability Structure for 32-bit Message Address and PVM



*Figure 7-47 MSI Capability Structure for 64-bit Message Address and PVM*

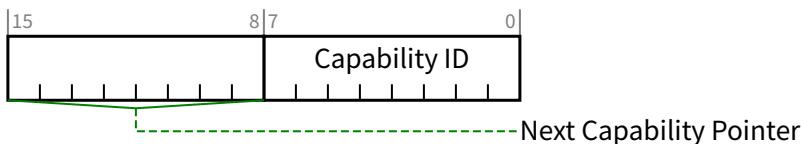
To request service, an MSI Function writes the contents of the Message Data Register for MSI, and if enabled, the Extended Message Data Register for MSI, to the address specified by the contents of the Message Address Register for MSI (and, optionally, when 64-bit message addresses are used, the Message Upper Address Register for MSI). A read of the address specified by the contents of the Message Address register produces undefined results.

A Function supporting MSI implements one of four MSI Capability structure layouts illustrated in Figure 7-44 to Figure 7-47, depending upon which optional features are supported. A Legacy Endpoint that implements MSI is required to support either the 32-bit or 64-bit Message Address version of the MSI Capability structure. A PCI Express Endpoint that implements MSI is required to support the 64-bit Message Address version of the MSI Capability structure. The Message Control Register for MSI indicates the Function's capabilities and provides system software control over MSI.

Each field is further described in the following sections.

### 7.7.1.1 MSI Capability Header (Offset 00h)

The MSI Capability Header enumerates the MSI Capability structure in the PCI Configuration Space Capability list. Figure 7-48 details allocation of register fields in the MSI Capability Header; Table 7-38 provides the respective bit definitions.



*Figure 7-48 MSI Capability Header*

Table 7-38 MSI Capability Header

Bit Location	Register Description	Attributes
7:0	<b>Capability ID</b> - Indicates the MSI Capability structure. This field must return a Capability ID of 05h indicating that this is an MSI Capability structure.	RO
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.	RO

### 7.7.1.2 Message Control Register for MSI (Offset 02h)

This register provides system software control over MSI. By default, MSI is disabled. If MSI and MSI-X are both disabled, the Function requests servicing using INTx interrupts (if supported). System software can enable MSI by Setting bit 0 of this register. System software is permitted to modify the Message Control Register for MSI's read-write bits and fields. A device driver is not permitted to modify the Message Control Register for MSI's read-write bits and fields.

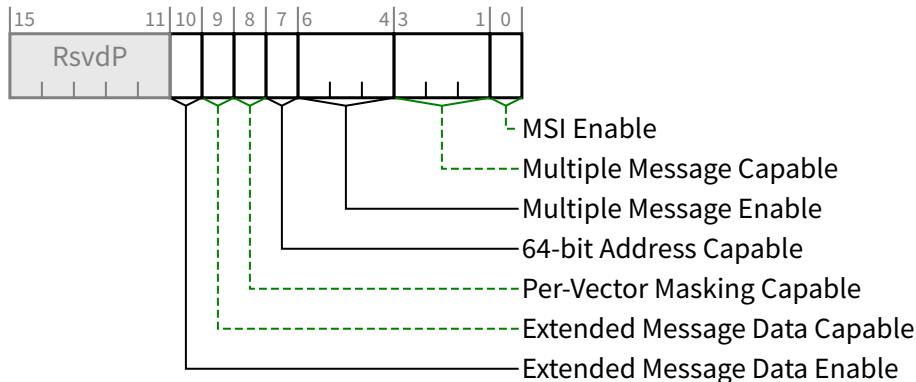


Figure 7-49 Message Control Register for MSI

Table 7-39 Message Control Register for MSI

Bit Location	Register Description	Attributes
0	<b>MSI Enable</b> - If Set and the <u>MSI-X Enable</u> bit in the <u>Message Control Register for MSI-X</u> (see <u>Section 7.9.2</u> ) is Clear, the Function is permitted to use MSI to request service and is prohibited from using INTx interrupts. System configuration software Sets this bit to enable MSI. A device driver is prohibited from writing this bit to mask a Function's service request. Refer to <u>Section 7.5.1.1</u> for control of INTx interrupts.  If Clear, the Function is prohibited from using MSI to request service.  Default value of this bit is 0b.	RW
3:1	<b>Multiple Message Capable</b> - System software reads this field to determine the number of requested vectors. The number of requested vectors must be aligned to a power of two (if a Function requires three vectors, it requests four by initializing this field to 010b). The encoding is defined as:  <b>000b</b> 1 vector requested <b>001b</b> 2 vectors requested	RO

Bit Location	Register Description	Attributes
	<p><b>010b</b> 4 vectors requested  <b>011b</b> 8 vectors requested  <b>100b</b> 16 vectors requested  <b>101b</b> 32 vectors requested  <b>110b</b> Reserved  <b>111b</b> Reserved</p>	
6:4	<p><b>Multiple Message Enable</b> - software writes to this field to indicate the number of allocated vectors (equal to or less than the number of requested vectors). The number of allocated vectors is aligned to a power of two. If a Function requests four vectors (indicated by a <u>Multiple Message Capable</u> encoding of 010b), system software can allocate either four, two, or one vector by writing a 010b, 001b, or 000b to this field, respectively. When <u>MSI Enable</u> is Set, a Function will be allocated at least 1 vector. The encoding is defined as:</p> <p><b>000b</b> 1 vector allocated  <b>001b</b> 2 vectors allocated  <b>010b</b> 4 vectors allocated  <b>011b</b> 8 vectors allocated  <b>100b</b> 16 vectors allocated  <b>101b</b> 32 vectors allocated  <b>110b</b> Reserved  <b>111b</b> Reserved</p> <p>Default value of this field is 000b.</p>	<u>RW</u>
7	<b>64-bit Address Capable</b> - If Set, the Function is capable of sending a 64-bit Message Address. If Clear, the Function is not capable of sending a 64-bit Message Address. This bit must be Set if the Function is a PCI Express Endpoint.	<u>RO</u>
8	<b>Per-Vector Masking Capable</b> - If Set, the Function supports MSI Per-Vector Masking. If Clear, the Function does not support MSI Per-Vector Masking. This bit must be Set if the Function is a PF or VF within an SR-IOV Device.	<u>RO</u>
9	<b>Extended Message Data Capable</b> - If Set, the Function is capable of providing <u>Extended Message Data</u> . If Clear, the Function does not support providing <u>Extended Message Data</u> .	<u>RO</u>
10	<p><b>Extended Message Data Enable</b> - If Set, the Function is enabled to provide <u>Extended Message Data</u>. If Clear, the Function is not enabled to provide <u>Extended Message Data</u>.</p> <p>Default value of this bit is 0b.</p> <p>This bit must be read-write if the <u>Extended Message Data Capable</u> bit is 1b; otherwise it must be hardwired to 0b.</p>	<u>RW/RO</u>

### 7.7.1.3 Message Address Register for MSI (Offset 04h)

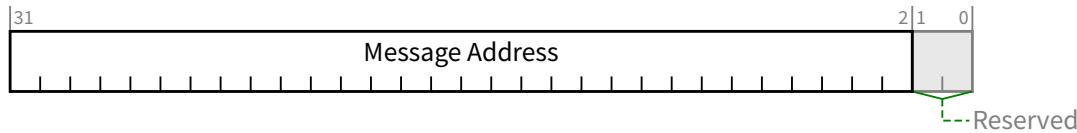


Figure 7-50 Message Address Register for MSI

Table 7-40 Message Address Register for MSI

Bit Location	Register Description	Attributes
1:0	<b>Reserved</b> - Always returns 0 on read. Write operations have no effect.	<u>RsvdP</u>
31:2	<b>Message Address</b> - System-specified message address. If the <u>MSI Enable</u> bit is Set, the contents of this register specify the DWORD-aligned address (Address[31:02]) for the MSI transaction. Address[1:0] are set to 00b. Default value of this field is undefined.	<u>RW</u>

### 7.7.1.4 Message Upper Address Register for MSI (Offset 08h)

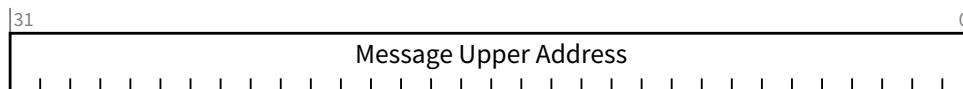


Figure 7-51 Message Upper Address Register for MSI

Table 7-41 Message Upper Address Register for MSI

Bit Location	Register Description	Attributes
31:0	<b>Message Upper Address</b> - System-specified message upper address. This register is implemented only if the Function supports a 64-bit message address ( <u>64-bit Address Capable</u> is Set). This register is required for PCI Express Endpoints and is optional for other Function types. If the <u>MSI Enable</u> bit is Set, the contents of this register (if non-zero) specify the upper 32-bits of a 64-bit message address (Address[63:32]). If the contents of this register are zero, the Function uses the 32 bit address specified by the Message Address register. Default value of this field is undefined.	<u>RW</u>

### 7.7.1.5 Message Data Register for MSI (Offset 08h or 0Ch)

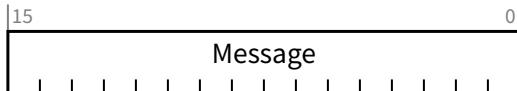


Figure 7-52 Message Data Register for MSI

Table 7-42 Message Data Register for MSI

Bit Location	Register Description	Attributes
15:0	<p><b>Message Data</b> - System-specified message data.</p> <p>If the <u>MSI Enable</u> bit is Set, the Function sends a DWORD Memory Write transaction using Message Data for the lower 16 bits. All 4 Byte Enables are Set.</p> <p>The <u>Multiple Message Enable</u> field defines the number of low order message data bits the Function is permitted to modify to generate its system software allocated vectors. For example, a <u>Multiple Message Enable</u> encoding of 010b indicates the Function has been allocated four vectors and is permitted to modify message data bits 1 and 0 (a Function modifies the lower message data bits to generate the allocated number of vectors). If the <u>Multiple Message Enable</u> field is 000b, the Function is not permitted to modify the message data.</p> <p>Default value of this field is undefined.</p>	<u>RW</u>

### 7.7.1.6 Extended Message Data Register for MSI (Optional)

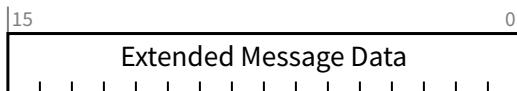


Figure 7-53 Extended Message Data Register for MSI

Table 7-43 Extended Message Data Register for MSI

Bit Location	Register Description	Attributes
15:0	<p><b>Extended Message Data</b> - System-specified message data.</p> <p>This register is optional. For the MSI Capability structures without Per-vector Masking, it must be implemented if the <u>Extended Message Data Capable</u> bit is Set; otherwise, it is outside the MSI Capability structure and undefined. For the MSI Capability structures with Per-vector Masking, it must be implemented if the <u>Extended Message Data Capable</u> bit is Set; otherwise, it is <u>RsvdP</u>.</p> <p>If the <u>Extended Message Data Enable</u> bit is Set, the DWORD Memory Write transaction uses <u>Extended Message Data</u> for the upper 16 bits; otherwise, it uses 0000h for the upper 16 bits.</p> <p>Default value of this field is 0000h.</p>	<u>RW/undefined/</u> <u>RsvdP</u>

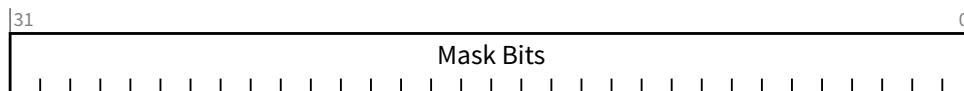
### 7.7.1.7 Mask Bits Register for MSI (Offset 0Ch or 10h)

This register is optional. It is present if Per-Vector Masking Capable is Set (see [Section 7.7.1.2](#)). The offset of this register within the capability depends on the value of the 64-bit Address Capable bit (see [Section 7.7.1.2](#)).

The Mask Bits and Pending Bits registers enable software to disable or defer message sending on a per-vector basis.

MSI vectors are numbered 0 through N-1, where N is the number of vectors allocated by software. Each vector is associated with a correspondingly numbered bit in the Mask Bits and Pending Bits registers.

The Multiple Message Capable field indicates how many vectors (with associated Mask and Pending bits) are implemented. All unimplemented Mask and Pending bits are Reserved.



*Figure 7-54 Mask Bits Register for MSI*

*Table 7-44 Mask Bits Register for MSI*

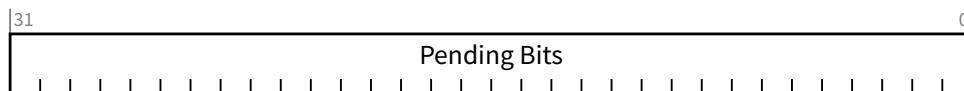
Bit Location	Register Description	Attributes
31:0	<b>Mask Bits</b> - For each Mask bit that is Set, the Function is prohibited from sending the associated message.  Default is 0.	<u>RW</u>

### 7.7.1.8 Pending Bits Register for MSI (Offset 10h or 14h)

This register is optional. It is present if Per-Vector Masking Capable is Set (see [Section 7.7.1.2](#)).

The offset of this register within the capability depends on the value of the 64-bit Address Capable bit (see [Section 7.7.1.2](#))

See [Section 7.7.1.7](#) for additional requirements on this register.



*Figure 7-55 Pending Bits Register for MSI*

*Table 7-45 Pending Bits Register for MSI*

Bit Location	Register Description	Attributes
31:0	<b>Pending Bits</b> - For each Pending bit that is Set, the Function has a pending associated message. Default is 0.	<u>RO</u>

## 7.7.2 MSI-X Capability and Table Structure

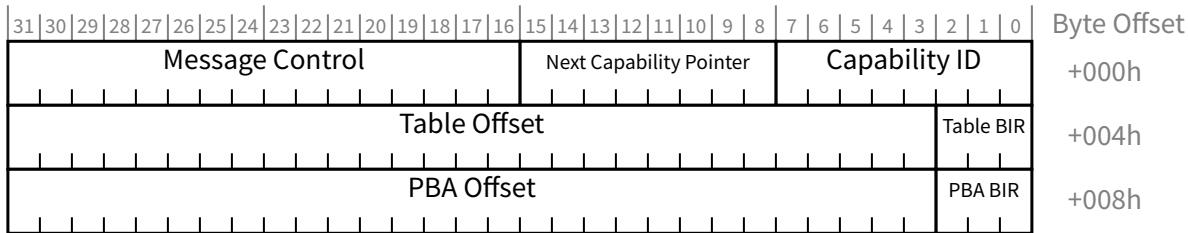
The MSI-X Capability structure is illustrated in Figure 7-56. More than one MSI-X Capability structure per Function is prohibited, but a Function is permitted to have both an MSI Capability structure and an MSI-X Capability structure.

In contrast to the MSI Capability structure, which directly contains all of the control/status information for the Function's vectors, the MSI-X Capability structure instead points to an **MSI-X Table** structure and an **MSI-X PBA** structure (Pending Bit Array structure), each residing in Memory Space (see Figure 7-57 and Figure 7-58 ).

Each structure is mapped by a Base Address Register (BAR) belonging to the Function, located beginning at 10h in Configuration Space, or entry in the Enhanced Allocation capability. A BAR Indicator register (BIR) indicates which BAR(or BEI when using Enhanced Allocation), and a QWORD-aligned Offset indicates where the structure begins relative to the base address associated with the BAR. The BAR is permitted to be either 32-bit or 64-bit, but must map Memory Space. A Function is permitted to map both structures with the same BAR, or to map each structure with a different BAR.

The MSI-X Table structure, illustrated in Figure 7-57, typically contains multiple entries, each consisting of several fields: Message Address, Message Upper Address, Message Data, and Vector Control. Each entry is capable of specifying a unique vector.

The Pending Bit Array (PBA) structure, illustrated in Figure 7-58, contains the Function's Pending Bits, one per Table entry, organized as a packed array of bits within QWORDs. The last QWORD will not necessarily be fully populated.

*Figure 7-56 MSI-X Capability Structure*

Byte Offset	
+000h	Entry 0: Message Address
+004h	Entry 0: Message Upper Address
+008h	Entry 0: Message Data
+00Ch	Entry 0: Vector Control
+010h	Entry 1: Message Address
+014h	Entry 1: Message Upper Address
+018h	Entry 1: Message Data
+01Ch	Entry 1: Vector Control
+020h	Entry 2: Message Address
+024h	Entry 2: Message Upper Address
+028h	Entry 2: Message Data
+02Ch	Entry 2: Vector Control
+030h	...

Figure 7-57 MSI-X Table Structure

Byte Offset	
+000h	Pending Bits 0 through 63
+004h	Pending Bits 64 through 127
+008h	...

Figure 7-58 MSI-X PBA Structure

To request service using a given MSI-X Table entry, a Function performs a DWORD Memory Write transaction using the contents of the Message Data field entry for data, the contents of the Message Upper Address field for the upper 32 bits of address, and the contents of the Message Address field entry for the lower 32 bits of address. A memory read transaction from the address targeted by the MSI-X message produces undefined results.

If a Base Address Register or entry in the Enhanced Allocation capability that maps address space for the MSI-X Table or MSI-X PBA also maps other usable address space that is not associated with MSI-X structures, locations (e.g., for CSRs) used in the other address space must not share any naturally aligned 4-KB address range with one where either MSI-X

structure resides. This allows system software where applicable to use different processor attributes for MSI-X structures and the other address space. (Some processor architectures do not support having different processor attributes associated with the same naturally aligned 4-KB physical address range.) The MSI-X Table and MSI-X PBA are permitted to co-reside within a naturally aligned 4-KB address range, though they must not overlap with each other.

## IMPLEMENTATION NOTE

### Dedicated BARs and Address Range Isolation

To enable system software to map MSI-X structures onto different processor pages for improved access control, it is recommended that a Function dedicate separate Base Address Registers for the MSI-X Table and MSI-X PBA, or else provide more than the minimum required isolation with address ranges.

If dedicated separate Base Address Registers is not feasible, it is recommended that a Function dedicate a single Base Address Register for the MSI-X Table and MSI-X PBA.

If a dedicated Base Address Register is not feasible, it is recommended that a Function isolate the MSI-X structures from the non-MSI-X structures with aligned 8 KB ranges rather than the mandatory aligned 4 KB ranges.

For example, if a Base Address Register needs to map 2 KB for an MSI-X Table containing 128 entries, 16 bytes for an MSI-X PBA containing 128 bits, and 64 bytes for registers not related to MSI-X, the following is an acceptable implementation. The Base Address Register requests 8 KB of total address space, maps the first 64 bytes for the non MSI-X registers, maps the MSI-X Table beginning at an offset of 4 KB, and maps the MSI-X PBA beginning at an offset of 6 KB.

A preferable implementation for a shared Base Address Register is for it to request 16 KB of total address space, map the first 64 bytes for the non MSI-X registers, map the MSI-X Table beginning at an offset of 8 KB, and map the MSI-X PBA beginning at an offset of 12 KB.

## IMPLEMENTATION NOTE

### MSI-X Memory Space Structures in Read/Write Memory

The MSI-X Table and MSI-X PBA structures are defined such that they can reside in general purpose read/write memory on a device, for ease of implementation and added flexibility. To achieve this, none of the contained fields are required to be read-only, and there are also restrictions on transaction alignment and sizes.

For all accesses to MSI-X Table and MSI-X PBA fields, software must use aligned full DWORD or aligned full QWORD transactions; otherwise, the result is undefined.

MSI-X Table entries and Pending bits are each numbered 0 through N-1, where N-1 is indicated by the Table Size field in the Message Control Register for MSI-X. For a given arbitrary MSI-X Table entry  $k$ , its starting address can be calculated with the formula:

$$\text{entry starting address} = \text{Table base} + k \times 16$$

*Equation 7-1 MSI-X Starting Address*

For the associated Pending bit  $k$ , its address for QWORD access and bit number within that QWORD can be calculated with the formulas:

$\text{QWORD address} = \text{PBA base} + (k \text{ div } 64) \times 8$

$\text{QWORD bit\#} = k \bmod 64$

Equation 7-2 MSI-X PBA QWORD Access

Software that chooses to read Pending bit  $K$  with DWORD accesses can use these formulas:

$\text{DWORD address} = \text{PBA base} + (k \text{ div } 32) \times 4$

$\text{DWORD bit\#} = k \bmod 32$

Equation 7-3 MSI-X PBA DWORD Access

Each field in the MSI-X Capability, MSI-X Table, and MSI-X PBA structures is further described in the following sections. Within the MSI-X Capability structure, Reserved registers and bits always return 0 when read, and write operations have no effect. Within the MSI-X Table and PBA structures, Reserved fields have special rules.

### 7.7.2.1 MSI-X Capability Header (Offset 00h)

The MSI-X Capability Header enumerates the MSI-X Capability structure in the PCI Configuration Space Capability list. Figure 7-56 details allocation of register fields in the MSI-X Capability Header; Table 7-46 provides the respective bit definitions.

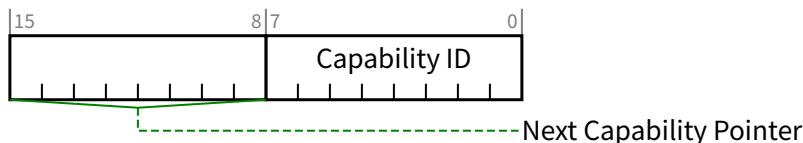


Figure 7-59 MSI-X Capability Header

Table 7-46 MSI-X Capability Header

Bit Location	Register Description	Attributes
7:0	<b>Capability ID</b> - Indicates the MSI-X Capability structure. This field must return a Capability ID of 11h indicating that this is an <u>MSI-X Capability</u> structure.	RO
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.	RO

### 7.7.2.2 Message Control Register for MSI-X (Offset 02h)

By default, MSI-X is disabled. If MSI and MSI-X are both disabled, the Function requests servicing via INTx interrupts (if supported). System software can enable MSI-X by Setting bit 15 of this register. System software is permitted to modify

the Message Control register's read-write bits and fields. A device driver is not permitted to modify the Message Control register's read-write bits and fields.

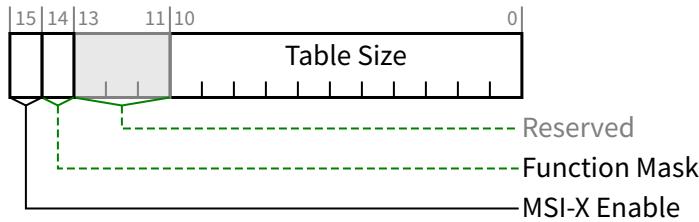


Figure 7-60 Message Control Register for MSI-X

Table 7-47 Message Control Register for MSI-X

Bit Location	Register Description	Attributes
10:0	<b>Table Size</b> - System software reads this field to determine the MSI-X Table Size N, which is encoded as N-1. For example, a returned value of 000 0000 0011b indicates a table size of 4.	<u>RO</u>
13:11	<b>Reserved</b> - Always returns 0 on a read, and a write operation has no effect.	<u>RsvdP</u>
14	<b>Function Mask</b> - If Set, all of the vectors associated with the Function are masked, regardless of their per-vector Mask bit values. If Clear, each vector's Mask bit determines whether the vector is masked or not. Setting or Clearing the MSI-X Function Mask bit has no effect on the value of the per-vector Mask bits. Default value of this bit is 0b.	<u>RW</u>
15	<b>MSI-X Enable</b> - If Set and the MSI Enable bit in the <a href="#">Message Control Register for MSI</a> (see <a href="#">Section 7.7.1.2</a> ) is Clear, the Function is permitted to use MSI-X to request service and is prohibited from using INTx interrupts (if implemented). System configuration software Sets this bit to enable MSI-X. A device driver is prohibited from writing this bit to mask a Function's service request. If Clear, the Function is prohibited from using MSI-X to request service. Default value of this bit is 0b.	<u>RW</u>

### 7.7.2.3 Table Offset/Table BIR Register for MSI-X (Offset 04h)

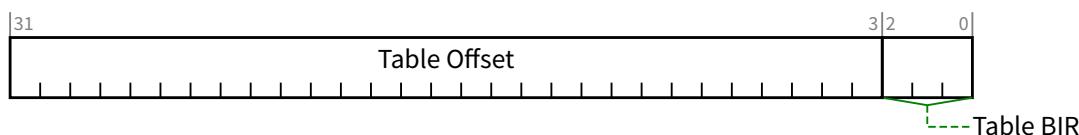


Figure 7-61 Table Offset/Table BIR Register for MSI-X

Table 7-48 Table Offset/Table BIR Register for MSI-X

Bit Location	Register Description	Attributes
2:0	<p><b>Table BIR</b> - Indicates which one of a Function's Base Address Registers, located beginning at 10h in Configuration Space, or entry in the Enhanced Allocation capability with a matching BAR Equivalent Indicator (BEI), is used to map the Function's MSI-X Table into Memory Space.</p> <p>Defined encodings are:</p> <ul style="list-style-type: none"> <li><b>0</b> Base Address Register 10h</li> <li><b>1</b> Base Address Register 14h</li> <li><b>2</b> Base Address Register 18h</li> <li><b>3</b> Base Address Register 1Ch</li> <li><b>4</b> Base Address Register 20h</li> <li><b>5</b> Base Address Register 24h</li> <li><b>6</b> Reserved</li> <li><b>7</b> Reserved</li> </ul> <p>For a 64-bit Base Address Register, the Table BIR indicates the lower DWORD. For Functions with Type 1 Configuration Space headers, BIR values 2 through 5 are also Reserved.</p>	RO
31:3	<p><b>Table Offset</b> - Used as an offset from the address contained by one of the Function's Base Address Registers to point to the base of the MSI-X Table. The lower 3 <u>Table BIR</u> bits are masked off (set to zero) by software to form a 32-bit QWORD-aligned offset.</p>	RO

#### 7.7.2.4 PBA Offset/PBA BIR Register for MSI-X (Offset 08h)

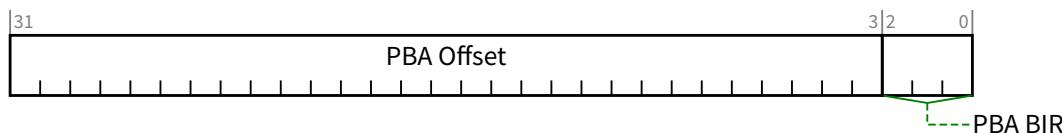


Figure 7-62 PBA Offset/PBA BIR Register for MSI-X

Table 7-49 PBA Offset/PBA BIR Register for MSI-X

Bit Location	Register Description	Attributes
2:0	<p><b>PBA BIR</b> - Indicates which one of a Function's Base Address Registers, located beginning at 10h in Configuration Space, or entry in the Enhanced Allocation capability with a matching BEI, is used to map the Function's MSI-X PBA into Memory Space.</p> <p>The <u>PBA BIR</u> value definitions are identical to those for the <u>Table BIR</u>.</p>	RO
31:3	<p><b>PBA Offset</b> - Used as an offset from the address contained by one of the Function's Base Address Registers to point to the base of the MSI-X PBA. The lower 3 <u>PBA BIR</u> bits are masked off (set to zero) by software to form a 32-bit QWORD-aligned offset.</p>	RO

### 7.7.2.5 Message Address Register for MSI-X Table Entries

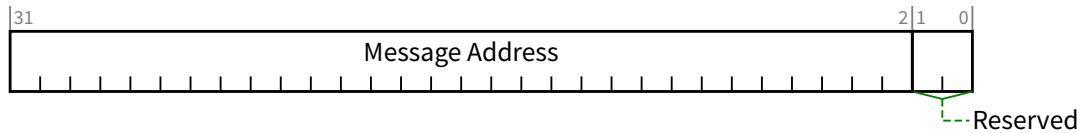


Figure 7-63 Message Address Register for MSI-X Table Entries

Table 7-50 Message Address Register for MSI-X Table Entries

Bit Location	Register Description	Attributes
1:0	<b>Reserved</b> - For proper DWORD alignment, software must always write zeroes to these two bits; otherwise the result is undefined. Default value of this field is 00b. These bits are permitted to be read-only or read-write.	<u>RO</u> or <u>RW</u>
31:2	<b>Message Address</b> - System-specified message lower address. For MSI-X messages, the contents of this field from an <u>MSI-X Table</u> entry specifies the lower portion of the DWORD-aligned address for the Memory Write transaction. Default value of this field is undefined.	<u>RW</u>

### 7.7.2.6 Message Upper Address Register for MSI-X Table Entries

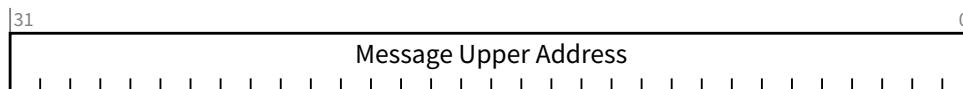


Figure 7-64 Message Upper Address Register for MSI-X Table Entries

Table 7-51 Message Upper Address Register for MSI-X Table Entries

Bit Location	Register Description	Attributes
31:0	<b>Message Upper Address</b> - System-specified message upper address bits. If this field is zero, 32-bit address messages are used. If this field is non-zero, 64-bit address messages are used. Default value of this field is undefined.	<u>RW</u>

### 7.7.2.7 Message Data Register for MSI-X Table Entries

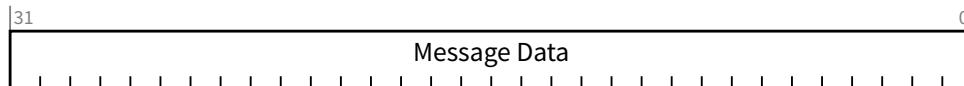


Figure 7-65 Message Data Register for MSI-X Table Entries

Table 7-52 Message Data Register for MSI-X Table Entries

Bit Location	Register Description	Attributes
31:0	<p><b>Message Data</b> - System-specified message data.</p> <p>For MSI-X messages, the contents of this field from an <u>MSI-X Table</u> entry specifies the 32-bit data payload of the DWORD Memory Write transaction. All 4 Byte Enables are Set.</p> <p>In contrast to message data used for MSI messages, the low-order message data bits in MSI-X messages are not modified by the Function.</p> <p>This field is read-write.</p> <p>Default value of this field is undefined.</p>	RW

### 7.7.2.8 Vector Control Register for MSI-X Table Entries

If a Function implements a TPH Requester Extended Capability structure and an MSI-X Capability structure, the Function can optionally use the Vector Control Register for MSI-X Table Entries in each entry to store a Steering Tag. See Section 6.17.

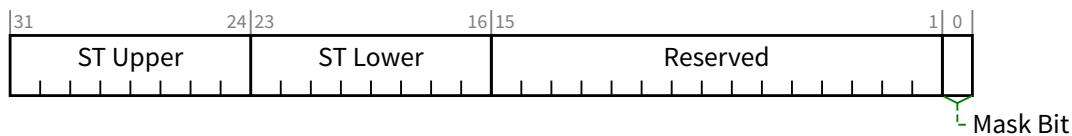


Figure 7-66 Vector Control Register for MSI-X Table Entries

Table 7-53 Vector Control Register for MSI-X Table Entries

Bit Location	Register Description	Attributes
0	<p><b>Mask Bit</b> - When this bit is Set, the Function is prohibited from sending a message using this <u>MSI-X Table</u> entry. However, any other <u>MSI-X Table</u> entries programmed with the same vector will still be capable of sending an equivalent message unless they are also masked.</p> <p>Default value of this bit is 1b (entry is masked)</p>	RW

Bit Location	Register Description	Attributes
15:1	<p><b>Reserved</b> - By default, the value of these bits must be 0. However, for potential future use, software must preserve the value of these Reserved bits when modifying the value of other Vector Control bits. If software modifies the value of these Reserved bits, the result is undefined.</p> <p>These bits are permitted to be <u>RsvdP</u> or read-write.</p>	<u>RW</u> or <u>RsvdP</u>
23:16	<p><b>ST Lower</b> - If the Function implements a TPH Requester Extended Capability structure, and the ST Table Location indicates a value of 10b, then this field contains the lower 8 bits of a Steering Tag and must be read-write.</p> <p>Otherwise, this field is permitted to be read-write or <u>RsvdP</u>, and for potential future use, software must preserve the value of these Reserved bits when modifying the value of other Vector Control bits, or the result is undefined.</p> <p>Default value of this field is 00h.</p>	<u>RW/RsvdP</u>
31:24	<p><b>ST Upper</b> - If the Function implements a TPH Requester Extended Capability structure, and the ST Table Location indicates a value of 10b, and the <u>Extended TPH Requester Supported</u> bit is Set, then this field contains the upper 8 bits of a Steering Tag and must be read-write.</p> <p>Otherwise, this field is permitted to be read-write or <u>RsvdP</u>, and for potential future use, software must preserve the value of these Reserved bits when modifying the value of other Vector Control bits, or the result is undefined.</p> <p>Default value of this field is 00h.</p>	<u>RW/RsvdP</u>

### 7.7.2.9 Pending Bits Register for MSI-X PBA Entries

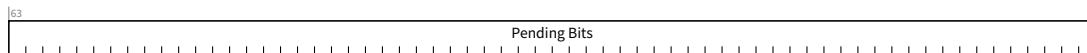


Figure 7-67 Pending Bits Register for MSI-X PBA Entries

Table 7-54 Pending Bits Register for MSI-X PBA Entries

Bit Location	Register Description	Attributes
63:0	<p><b>Pending Bits</b> - For each Pending Bit that is Set, the Function has a pending message for the associated MSI-X Table entry.</p> <p>Pending bits that have no associated <u>MSI-X Table</u> entry are Reserved. By default, the value of Reserved Pending bits must be 0b.</p> <p>Software should never write, and should only read Pending Bits. If software writes to Pending Bits, the result is undefined.</p> <p>Default value of each Pending Bit is 0b.</p> <p>These bits are permitted to be read-only or read-write.</p>	<u>RO</u> or <u>RW</u>

### 7.7.3 Secondary PCI Express Extended Capability

The Secondary PCI Express Extended Capability structure must be implemented in any Function or RCRB where any of the following are true:

- The Supported Link Speeds Vector field indicates that the Link supports Link Speeds of 8.0 GT/s or higher (see Section 7.5.3.18 or Section 7.9.9.2 ).
- Any bit in the Lower SKP OS Generation Supported Speeds Vector field is Set (see Section 7.5.3.18 ).
- When Lane based errors are reported in the Lane Error Status register (discussed in Section 4.2.6 ).

To support future additions to this capability, this capability is permitted in any Function or RCRB associated with a Link. For a Multi-Function Device associated with an Upstream Port, this capability is permitted only in Function 0 of the Device.

Byte Offset	
+000h	PCI Express Extended Capability Header
+004h	Link Control 3 Register
+008h	Lane Error Status Register
+00Ch	Lane (1) Equalization Control Register Entry
+010h	Lane (3) Equalization Control Register Entry
+014h	Lane (5) Equalization Control Register Entry
+018h	Lane (7) Equalization Control Register Entry
+01Ch	Lane (9) Equalization Control Register Entry
+020h	Lane (11) Equalization Control Register Entry
+024h	Lane (13) Equalization Control Register Entry
+028h	Lane (15) Equalization Control Register Entry
+02Ch	Lane (17) Equalization Control Register Entry
+030h	Lane (19) Equalization Control Register Entry
+034h	Lane (21) Equalization Control Register Entry
+038h	Lane (23) Equalization Control Register Entry
+03Ch	Lane (25) Equalization Control Register Entry
+040h	Lane (27) Equalization Control Register Entry
+044h	Lane (29) Equalization Control Register Entry
+048h	Lane (31) Equalization Control Register Entry
Lane (0) Equalization Control Register Entry	
Lane (2) Equalization Control Register Entry	
Lane (4) Equalization Control Register Entry	
Lane (6) Equalization Control Register Entry	
Lane (8) Equalization Control Register Entry	
Lane (10) Equalization Control Register Entry	
Lane (12) Equalization Control Register Entry	
Lane (14) Equalization Control Register Entry	
Lane (16) Equalization Control Register Entry	
Lane (18) Equalization Control Register Entry	
Lane (20) Equalization Control Register Entry	
Lane (22) Equalization Control Register Entry	
Lane (24) Equalization Control Register Entry	
Lane (26) Equalization Control Register Entry	
Lane (28) Equalization Control Register Entry	
Lane (30) Equalization Control Register Entry	

*Figure 7-68 Secondary PCI Express Extended Capability Structure*

### 7.7.3.1 Secondary PCI Express Extended Capability Header (Offset 00h)

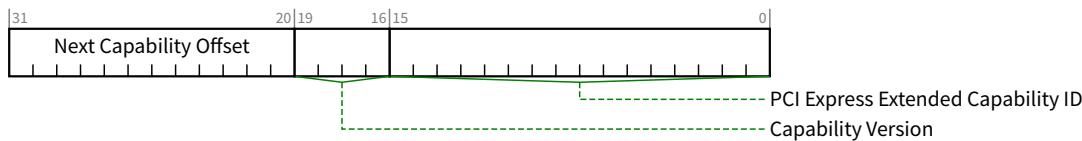


Figure 7-69 Secondary PCI Express Extended Capability Header

Table 7-55 Secondary PCI Express 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 Secondary PCI Express Extended Capability is 0019h.	<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>

### 7.7.3.2 Link Control 3 Register (Offset 04h)

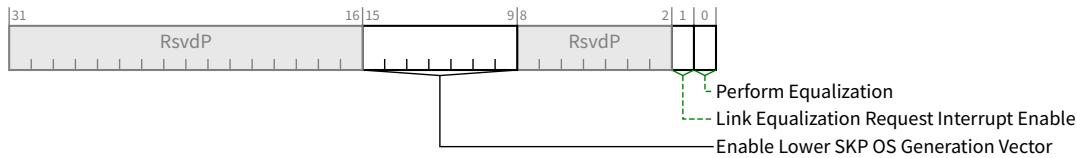


Figure 7-70 Link Control 3 Register

Table 7-56 Link Control 3 Register

Bit Location	Register Description	Attributes
0	<b>Perform Equalization</b> - When this bit is 1b and a 1b is written to the Retrain Link bit with the Target Link Speed field set to 8.0 GT/s or higher, the Downstream Port must perform Link Equalization. Refer to Section 4.2.3 and Section 4.2.6.4.2 for details.  This bit is RW for Downstream Ports and for Upstream Ports when Crosslink Supported is 1b (see Section 7.5.3.18). This bit is not applicable and is RsvdP for Upstream Ports when the Crosslink Supported bit is 0b.  The default value is 0b.  If the Port does not support 8.0 GT/s, this bit is permitted to be hardwired to 0b.	<u>RW/RsvdP</u>

Bit Location	Register Description	Attributes												
1	<p><b>Link Equalization Request Interrupt Enable</b> - When Set, this bit enables the generation of an interrupt to indicate that the <u>Link Equalization Request 8.0 GT/s</u> bit, the <u>Link Equalization Request 16.0 GT/s</u> bit, or the <u>Link Equalization Request 32.0 GT/s</u> bit has been set.</p> <p>This bit is RW for Downstream Ports and for Upstream Ports when <u>Crosslink Supported</u> is 1b (see Section 7.5.3.18). This bit is not applicable and is <u>RsvdP</u> for Upstream Ports when the <u>Crosslink Supported</u> bit is 0b.</p> <p>The default value for this bit is 0b.</p> <p>If the Port does not support 8.0 GT/s, this bit is permitted to be hardwired to 0b.</p>	RW/RsvdP												
9:15	<p><b>Enable Lower SKP OS Generation Vector</b> - When the Link is in L0 and the bit in this field corresponding to the <u>Current Link Speed</u> is Set, SKP Ordered Sets are scheduled at the rate defined for <u>SRNS</u>, overriding the rate required based on the clock tolerance architecture. See Section 4.2.7 for additional requirements.</p> <p>Bit definitions within this field are:</p> <table> <tbody> <tr> <td><b>Bit 0</b></td> <td>2.5 GT/s</td> </tr> <tr> <td><b>Bit 1</b></td> <td>5.0 GT/s</td> </tr> <tr> <td><b>Bit 2</b></td> <td>8.0 GT/s</td> </tr> <tr> <td><b>Bit 3</b></td> <td>16.0 GT/s</td> </tr> <tr> <td><b>Bit 4</b></td> <td>32.0 GT/s</td> </tr> <tr> <td><b>Bits 6:5</b></td> <td>RsvdP</td> </tr> </tbody> </table> <p>Each unreserved bit in this field must be RW if the corresponding bit in the <u>Lower SKP OS Generation Supported Speeds Vector</u> is Set, otherwise the bit must be RW or hardwired to 0.</p> <p>Behavior is undefined if a bit is Set in this field and the corresponding bit in the <u>Lower SKP OS Generation Supported Speeds Vector</u> is not Set.</p> <p>The default value of this field is 000 0000b.</p>	<b>Bit 0</b>	2.5 GT/s	<b>Bit 1</b>	5.0 GT/s	<b>Bit 2</b>	8.0 GT/s	<b>Bit 3</b>	16.0 GT/s	<b>Bit 4</b>	32.0 GT/s	<b>Bits 6:5</b>	RsvdP	RW/RsvdP
<b>Bit 0</b>	2.5 GT/s													
<b>Bit 1</b>	5.0 GT/s													
<b>Bit 2</b>	8.0 GT/s													
<b>Bit 3</b>	16.0 GT/s													
<b>Bit 4</b>	32.0 GT/s													
<b>Bits 6:5</b>	RsvdP													

### 7.7.3.3 Lane Error Status Register (Offset 08h)

The Lane Error Status Register consists of a 32-bit vector, where each bit indicates if the Lane with the corresponding Lane number detected an error. This Lane number is the default Lane number which is invariant to Link width and Lane reversal negotiation that occurs during Link training.

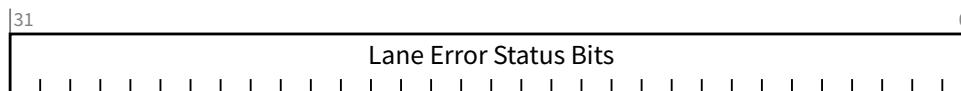


Figure 7-71 Lane Error Status Register

*Table 7-57 Lane Error Status Register*

Bit Location	Register Description	Attributes
31:0	<p><b>Lane Error Status Bits</b> - Each bit indicates if the corresponding Lane detected a Lane-based error. A value of 1b indicates that a Lane based-error was detected on the corresponding Lane Number (see <a href="#">Section 4.2.2.3.3</a>, <a href="#">Section 4.2.6</a>, and <a href="#">Section 4.2.7.2</a> for details).</p> <p>The default value of each bit is 0b.</p> <p>For Ports that are narrower than 32 Lanes, the unused upper bits [31: <a href="#">Maximum Link Width</a>] are <a href="#">RsvdZ</a>.</p> <p>For Ports that do not support 8.0 GT/s and do not set these bits based on 8b/10b errors (optional, see <a href="#">Section 4.2.6</a>), this field is permitted to be hardwired to 0.</p>	RW1CS

#### 7.7.3.4 Lane Equalization Control Register (Offset 0Ch)

The [Lane Equalization Control Register](#) consists of control fields required for per-Lane 8.0 GT/s equalization and the number of entries in this register are sized by [Maximum Link Width](#) (see [Section 7.5.3.6](#)). Each entry contains the values for the Lane with the corresponding default Lane number which is invariant to Link width and Lane reversal negotiation that occurs during Link training.

If the Port does not support 8.0 GT/s, this register is permitted to be hardwired to 0.

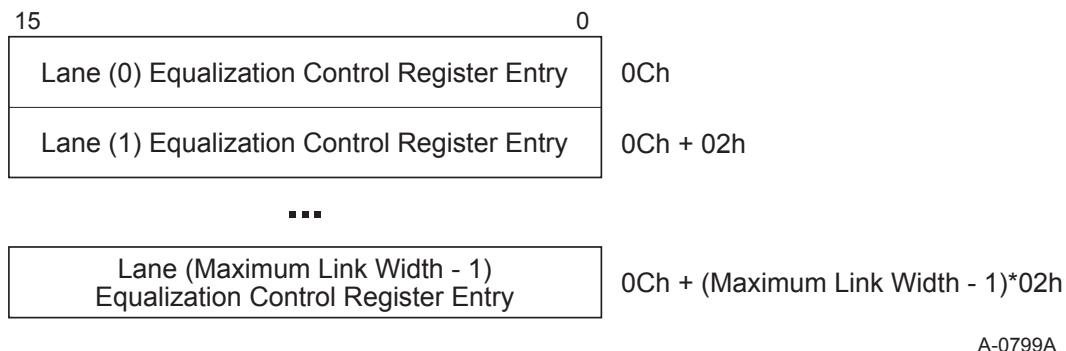
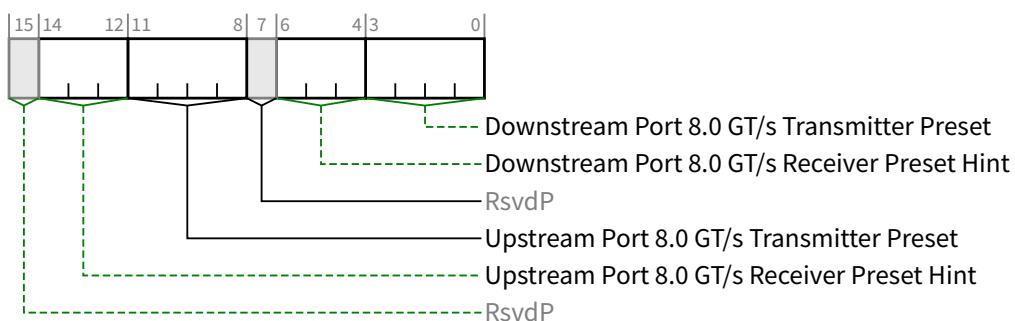
*Figure 7-72 Lane Equalization Control Register**Figure 7-73 Lane Equalization Control Register Entry*

Table 7-58 Lane Equalization Control Register Entry

Bit Location	Register Description	Attributes																
3:0	<p><b>Downstream Port 8.0 GT/s Transmitter Preset</b> - Transmitter preset value used for 8.0 GT/s equalization by this Port when the Port is operating as a Downstream Port. This field is ignored when the Port is operating as an Upstream Port. See <a href="#">Chapter 8</a> for details. The field encodings are defined in <a href="#">Section 4.2.3.2</a>.</p> <p>For an Upstream Port if <u>Crosslink Supported</u> is 0b, this field is <u>RsvdP</u>. Otherwise, this field is <u>HwInit</u>. See <a href="#">Section 7.5.3.18</a>.</p> <p>The default value is 111b.</p>	<u>HwInit/RsvdP</u> (see description)																
6:4	<p><b>Downstream Port 8.0 GT/s Receiver Preset Hint</b> - Receiver preset hint value that may be used as a suggested setting for 8.0 GT/s receiver equalization by this Port when the Port is operating as a Downstream Port. This field is ignored when the Port is operating as an Upstream Port. See <a href="#">Chapter 8</a> for details. The field encodings are defined in <a href="#">Section 4.2.3.2</a>.</p> <p>For an Upstream Port if <u>Crosslink Supported</u> is 0b, this field is <u>RsvdP</u>. Otherwise, this field is <u>HwInit</u>. See <a href="#">Section 7.5.3.18</a>.</p> <p>The default value is 11b.</p>	<u>HwInit/RsvdP</u> (see description)																
11:8	<p><b>Upstream Port 8.0 GT/s Transmitter Preset</b> - Field contains the Transmitter preset value sent or received during 8.0 GT/s Link Equalization. Field usage varies as follows:</p> <table border="1"> <thead> <tr> <th></th> <th>Operating Port Direction</th> <th><u>Crosslink Supported</u></th> <th>Usage</th> </tr> </thead> <tbody> <tr> <td>A</td> <td>Downstream Port</td> <td>Any</td> <td> <p>Field contains the value sent on the associated Lane during Link Equalization.</p> <p>Field is <u>HwInit</u>.</p> </td></tr> <tr> <td>B</td> <td>Upstream Port</td> <td>0b</td> <td> <p>Field is intended for debug and diagnostics. It contains the value captured from the associated Lane during Link Equalization.</p> <p>Field is RO.</p> <p>Note: When crosslinks are supported, case C (below) applies and this captured information is not visible to software. Vendors are encouraged to provide an alternate mechanism to obtain this information.</p> </td></tr> <tr> <td>C</td> <td>Upstream Port</td> <td>1b</td> <td> <p>Field is not used or affected by the current Link Equalization.</p> <p>Field value will be used if a future crosslink negotiation switches the Operating Port Direction so that case A (above) applies.</p> <p>Field is <u>HwInit</u>.</p> </td></tr> </tbody> </table> <p>See <a href="#">Section 4.2.3</a> and <a href="#">Chapter 8</a> for details. The field encodings are defined in <a href="#">Section 4.2.3.2</a>.</p> <p>The default value is 111b.</p>		Operating Port Direction	<u>Crosslink Supported</u>	Usage	A	Downstream Port	Any	<p>Field contains the value sent on the associated Lane during Link Equalization.</p> <p>Field is <u>HwInit</u>.</p>	B	Upstream Port	0b	<p>Field is intended for debug and diagnostics. It contains the value captured from the associated Lane during Link Equalization.</p> <p>Field is RO.</p> <p>Note: When crosslinks are supported, case C (below) applies and this captured information is not visible to software. Vendors are encouraged to provide an alternate mechanism to obtain this information.</p>	C	Upstream Port	1b	<p>Field is not used or affected by the current Link Equalization.</p> <p>Field value will be used if a future crosslink negotiation switches the Operating Port Direction so that case A (above) applies.</p> <p>Field is <u>HwInit</u>.</p>	<u>HwInit/RO</u> (see description)
	Operating Port Direction	<u>Crosslink Supported</u>	Usage															
A	Downstream Port	Any	<p>Field contains the value sent on the associated Lane during Link Equalization.</p> <p>Field is <u>HwInit</u>.</p>															
B	Upstream Port	0b	<p>Field is intended for debug and diagnostics. It contains the value captured from the associated Lane during Link Equalization.</p> <p>Field is RO.</p> <p>Note: When crosslinks are supported, case C (below) applies and this captured information is not visible to software. Vendors are encouraged to provide an alternate mechanism to obtain this information.</p>															
C	Upstream Port	1b	<p>Field is not used or affected by the current Link Equalization.</p> <p>Field value will be used if a future crosslink negotiation switches the Operating Port Direction so that case A (above) applies.</p> <p>Field is <u>HwInit</u>.</p>															
14:12	<p><b>Upstream Port 8.0 GT/s Receiver Preset Hint</b> - Field contains the Receiver preset hint value sent or received during 8.0 GT/s Link Equalization. Field usage varies as follows:</p>	<u>HwInit/RO</u> (see description)																

Bit Location	Register Description			Attributes
		Operating Port Direction	Crosslink Supported	
A	Downstream Port	Any		<p>Field contains the value sent on the associated Lane during Link Equalization.</p> <p>Field is <u>HwInit</u>.</p>
B	Upstream Port	0b		<p>Field is intended for debug and diagnostics. It contains the value captured from the associated Lane during Link Equalization.</p> <p>Field is <u>RO</u>.</p> <p>Note: When crosslinks are supported, case C (below) applies and this captured information is not visible to software. Vendors are encouraged to provide an alternate mechanism to obtain this information.</p>
C	Upstream Port	1b		<p>Field is not used or affected by the current Link Equalization.</p> <p>Field value will be used if a future crosslink negotiation switches the Operating Port Direction so that case A (above) applies.</p> <p>Field is <u>HwInit</u>.</p>
<p>See <a href="#">Section 4.2.3</a> and <a href="#">Chapter 8</a> for details. The field encodings are defined in <a href="#">Section 4.2.3.2</a>.</p> <p>The default value is 111b.</p>				

#### 7.7.4 Data Link Feature Extended Capability

The Data Link Feature Capability is an optional Extended Capability that is required for Downstream Ports that support one or more of the associated features. Since the Scaled Flow Control Feature is required for Ports that support 16.0 GT/s, this capability is required for Downstream Ports that support 16.0 GT/s (see [Section 3.4.2](#)). It is optional in other Downstream Ports. It is optional in Functions associated with an Upstream Port. In Multi-Function Devices associated with an Upstream Port, all instances of this capability must report identical information in all fields of this capability. It is not applicable in Functions that are not associated with a Port (e.g., RCiEPs, Root Complex Event Collectors). The Data Link Feature Extended Capability is shown in [Figure 7-74](#).

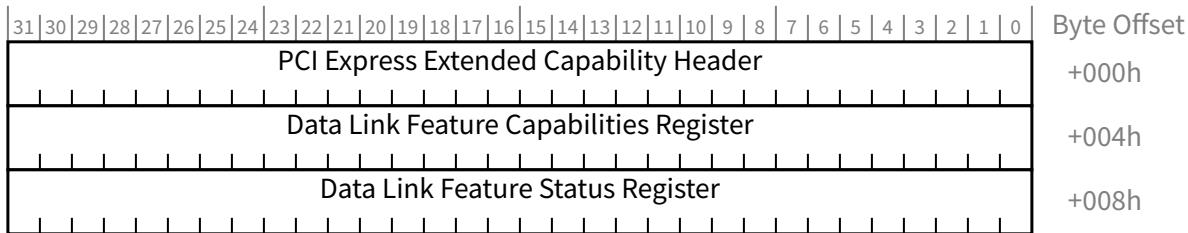


Figure 7-74 Data Link Feature Extended Capability

#### 7.7.4.1 Data Link Feature Extended Capability Header (Offset 00h)

Figure 7-75 details allocation of register fields in the Data Link Feature Extended Capability Header; Table 7-59 provides the respective bit definitions.

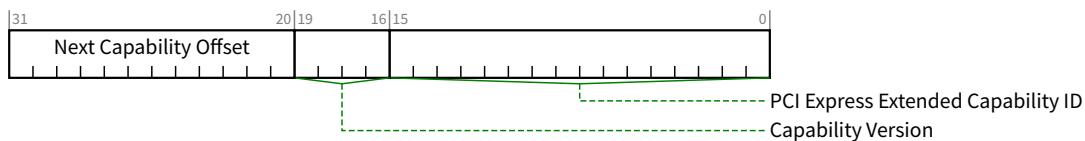


Figure 7-75 Data Link Feature Extended Capability Header

Table 7-59 Data Link Feature Extended Capability Header

Bit Location	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 Data Link Feature is 0025h	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 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.	RO

### 7.7.4.2 Data Link Feature Capabilities Register (Offset 04h)

Figure 7-76 details allocation of register fields in the Data Link Feature Capabilities register; Table 7-60 provides the respective bit definitions.

When this Port sends a Data Link Feature DLLP, the Feature Support field in Symbols 1, 2, and 3 of that DLLP contains bits [22:16], [15:8], and [7:0] of this register respectively (See Figure 3-12).

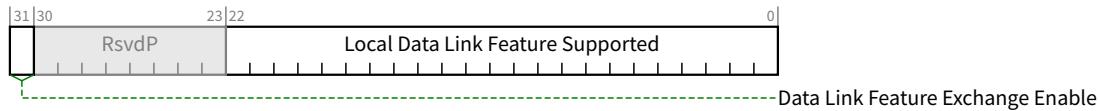


Figure 7-76 Data Link Feature Capabilities Register

Table 7-60 Data Link Feature Capabilities Register

Bit Location	Description	Attributes
22:0	<p><b>Local Data Link Feature Supported</b> - This field contains the Feature Supported value used when this Port sends a Data Link Feature DLLP (see Figure 3-12). Defined features are:</p> <ul style="list-style-type: none"> <li><b>Bit 0</b>    <b>Local Scaled Flow Control Supported</b> - This bit indicates that this Port supports the Scaled Flow Control Feature (see Section 3.4.2).</li> <li><b>Bits 22:1</b>    RsvdP</li> </ul> <p>Bits associated with features that this Port is capable of supporting are <u>HwInit</u>, defaulting to 1b. Other bits in this field are RsvdP.</p>	<u>HwInit/RsvdP</u>
31	<p><b>Data Link Feature Exchange Enable</b> - If Set, this bit indicates that this Port will enter the <u>DL_Feature</u> negotiation state (see Section 3.2.1). Default is 1b.</p>	<u>HwInit</u>

### 7.7.4.3 Data Link Feature Status Register (Offset 08h)

Figure 7-77 details allocation of register fields in the Data Link Feature Status Register; Table 7-61 provides the respective bit definitions.

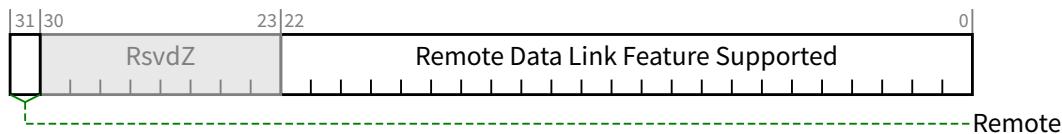


Figure 7-77 Data Link Feature Status Register

*Table 7-61 Data Link Feature Status Register*

Bit Location	Description	Attributes
22:0	<p><b>Remote Data Link Feature Supported</b> - These bits indicate that the Remote Port supports the corresponding Data Link Feature. These bits capture all information from the Feature Supported field of the <u>Data Link Feature DLLP</u> even when this Port doesn't support the corresponding feature.</p> <p>This field is Cleared on entry to state <u>DL_Inactive</u> (see <u>Section 3.2.1</u> ).</p> <p>Features currently defined are:</p> <ul style="list-style-type: none"> <li><b>Bit 0 -</b> This bit indicates that the Remote Port supports the <u>Scaled Flow Control Feature</u> (see <b>Remote</b> <u>Section 3.4.2</u> ).</li> <li><b>Scaled Flow Control Bits 22:1</b> Undefined</li> <li><b>Supported</b> Default is 00 0000h</li> </ul>	<u>RO</u>
31	<p><b>Remote Data Link Feature Supported Valid</b> - This bit indicates that the Port has received a <u>Data Link Feature DLLP</u> in state <u>DL_Feature</u> (see <u>Section 3.2.1</u> ) and that the Remote Data Link Feature Supported field is meaningful. This bit is Cleared on entry to state <u>DL_Inactive</u> (see <u>Section 3.2.1</u> ).</p> <p>Default is 0b.</p>	<u>RO</u>

### 7.7.5 Physical Layer 16.0 GT/s Extended Capability

The Physical Layer 16.0 GT/s Extended Capability structure must be implemented in:

- A Function associated with a Downstream Port where the Supported Link Speeds Vector field indicates support for a Link speed of 16.0 GT/s.
- A Function of a single-Function Device associated with an Upstream Port where the Supported Link Speeds Vector field indicates support for a Link speed of 16.0 GT/s.
- Function 0 (and only Function 0) of a Multi-Function Device associated with an Upstream Port where the Supported Link Speeds Vector field indicates support for a Link speed of 16.0 GT/s.

This capability is permitted to be implemented in any of the Functions listed above even if the 16.0 GT/s Link speed is not supported. When the 16.0 GT/s Link speed is not supported, the behavior of registers other than the Capability Header is undefined.

Figure 7-79 details allocation of register fields in the Physical Layer 16.0 GT/s Extended Capability structure.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Byte Offset
PCI Express Extended Capability Header																																
16.0 GT/s Capabilities Register																																
16.0 GT/s Control Register																																
16.0 GT/s Status Register																																
16.0 GT/s Local Data Parity Mismatch Status Register																																
16.0 GT/s First Retimer Data Parity Mismatch Status Register																																
16.0 GT/s Second Retimer Data Parity Mismatch Status Register																																
16.0 GT/s Reserved																																
16.0 GT/s Eq Ctl: Lane 3							16.0 GT/s Eq Ctl: Lane 2							16.0 GT/s Eq Ctl: Lane 1							16.0 GT/s Eq Ctl: Lane 0											
16.0 GT/s Eq Ctl: Lane 7							16.0 GT/s Eq Ctl: Lane 6							16.0 GT/s Eq Ctl: Lane 5							16.0 GT/s Eq Ctl: Lane 4											
16.0 GT/s Eq Ctl: Lane 11							16.0 GT/s Eq Ctl: Lane 10							16.0 GT/s Eq Ctl: Lane 9							16.0 GT/s Eq Ctl: Lane 8											
16.0 GT/s Eq Ctl: Lane 15							16.0 GT/s Eq Ctl: Lane 14							16.0 GT/s Eq Ctl: Lane 13							16.0 GT/s Eq Ctl: Lane 12											
16.0 GT/s Eq Ctl: Lane 19							16.0 GT/s Eq Ctl: Lane 18							16.0 GT/s Eq Ctl: Lane 17							16.0 GT/s Eq Ctl: Lane 16											
16.0 GT/s Eq Ctl: Lane 23							16.0 GT/s Eq Ctl: Lane 22							16.0 GT/s Eq Ctl: Lane 21							16.0 GT/s Eq Ctl: Lane 20											
16.0 GT/s Eq Ctl: Lane 27							16.0 GT/s Eq Ctl: Lane 26							16.0 GT/s Eq Ctl: Lane 25							16.0 GT/s Eq Ctl: Lane 24											
16.0 GT/s Eq Ctl: Lane 31							16.0 GT/s Eq Ctl: Lane 30							16.0 GT/s Eq Ctl: Lane 29							16.0 GT/s Eq Ctl: Lane 28											

Figure 7-78 Physical Layer 16.0 GT/s Extended Capability

### 7.7.5.1 Physical Layer 16.0 GT/s Extended Capability Header (Offset 00h)

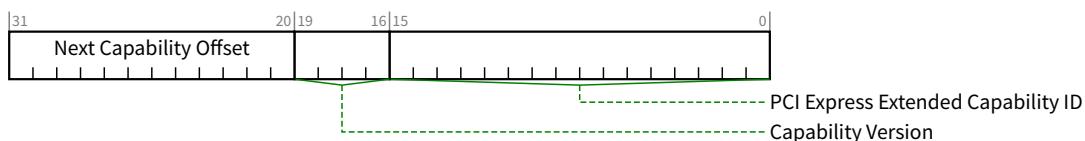


Figure 7-79 Physical Layer 16.0 GT/s Extended Capability Header