ATP Electronics, Inc.

NVME Overview

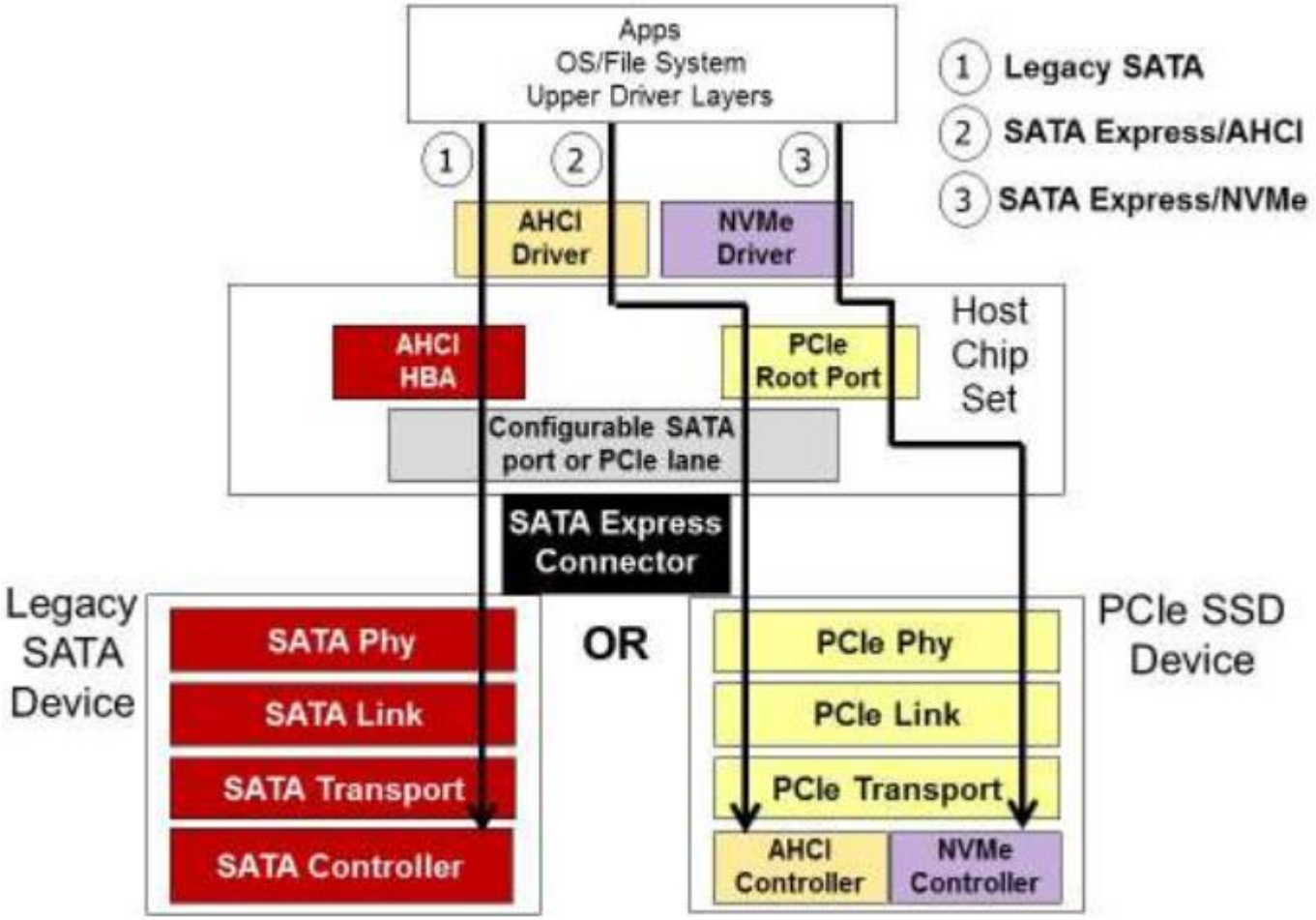
2021-03-16

Win Chen

Agenda

- AHCI VS NVME
- NVME Development
- NVME Attributes
- NVME Command
- Theory of Operation
- PRP & SGL
- Command Arbitration
- Interrupt
- Power Management
- End-to-end Data Protection
- Enterprise VS Client
- NVME NEW Feature

SATA Express Interface Architecture

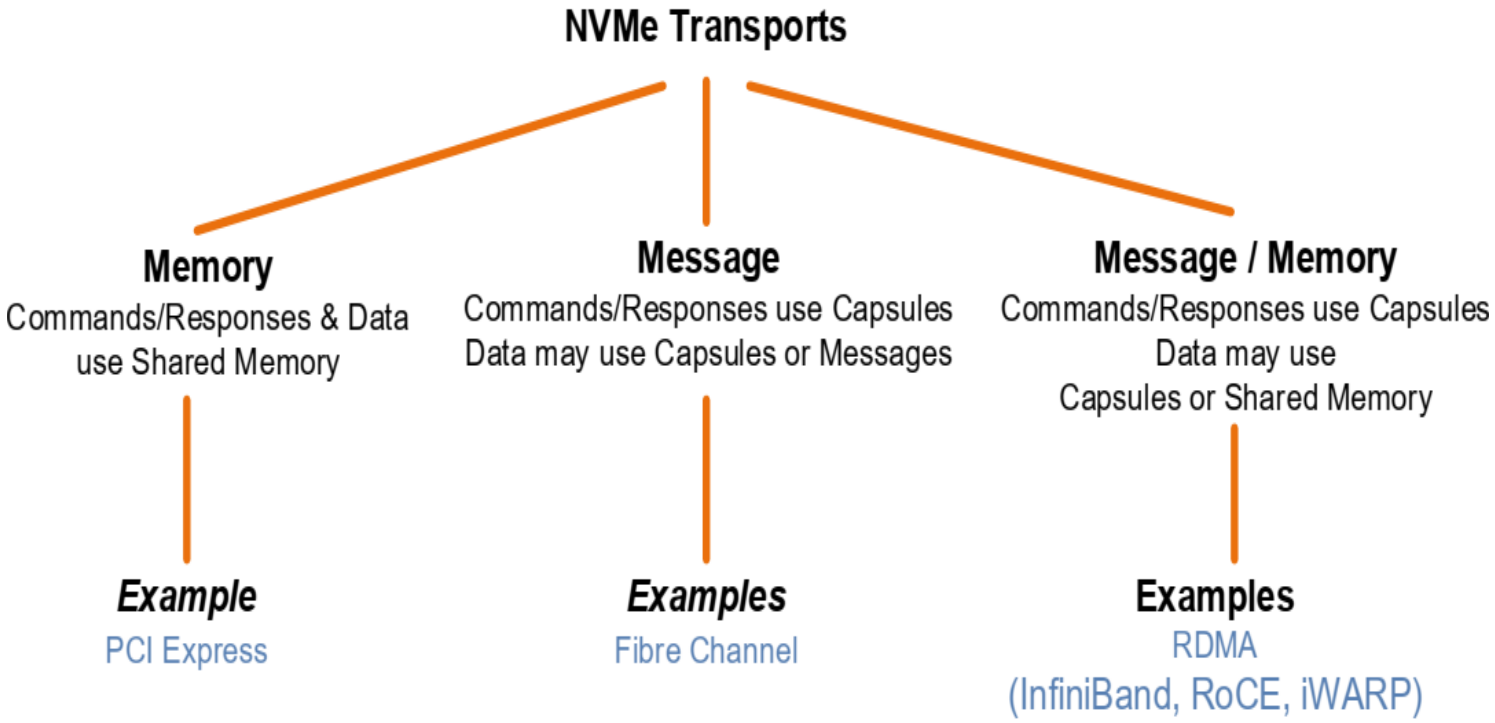


NVMe and AHCI Comparison

	AHCI	NVMe
Maximum Queue Depth	1 command queue 32 commands per Q	64K queues 64K Commands per Q
Un-cacheable register accesses (2K cycles each)	6 per non-queued command 9 per queued command	2 per command
MXI-X and Interrupt Steering	Single interrupt; no steering	2K MSI-X interrupts
Parallelism & Multiple Threads	Requires synchronization lock to issue command	No locking
Efficiency for 4KB Commands	Command parameters require two serialized host DRAM fetches	Command parameters in one 64B fetch

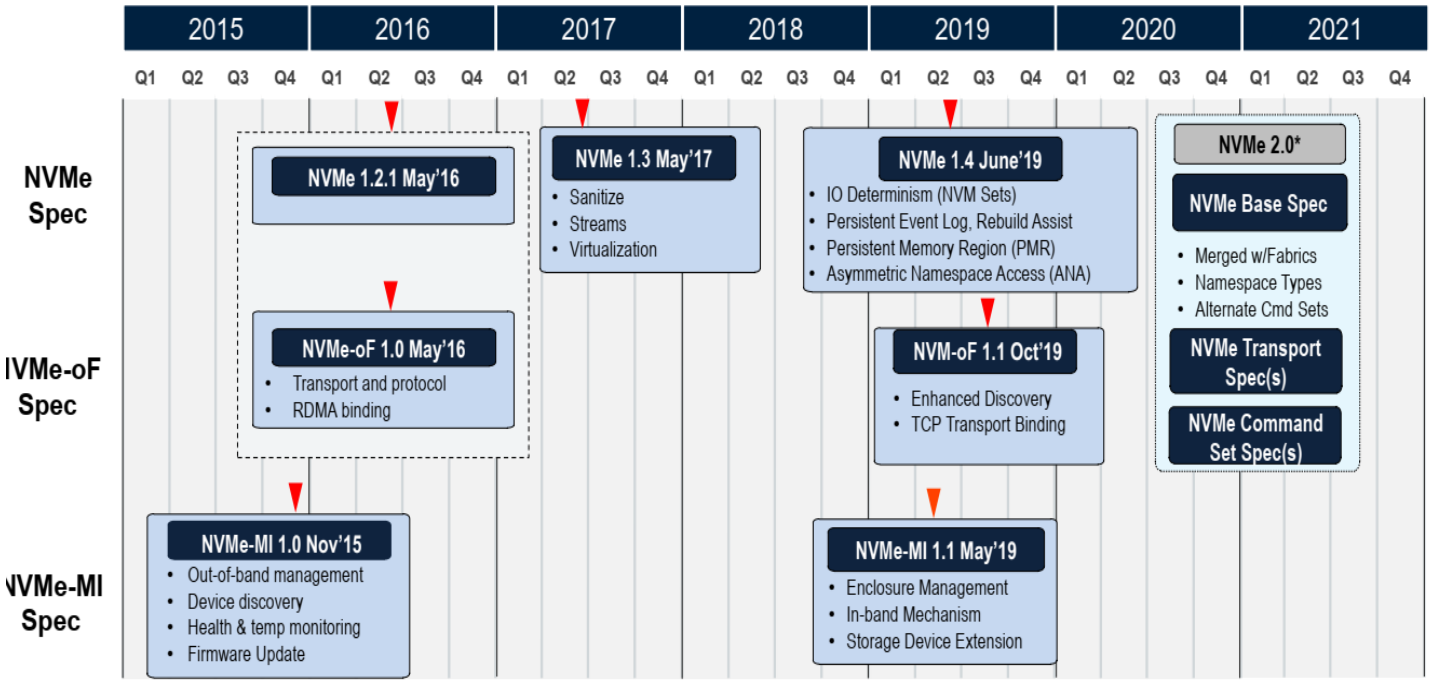
NVME Transports

Figure 1: Taxonomy of Transports

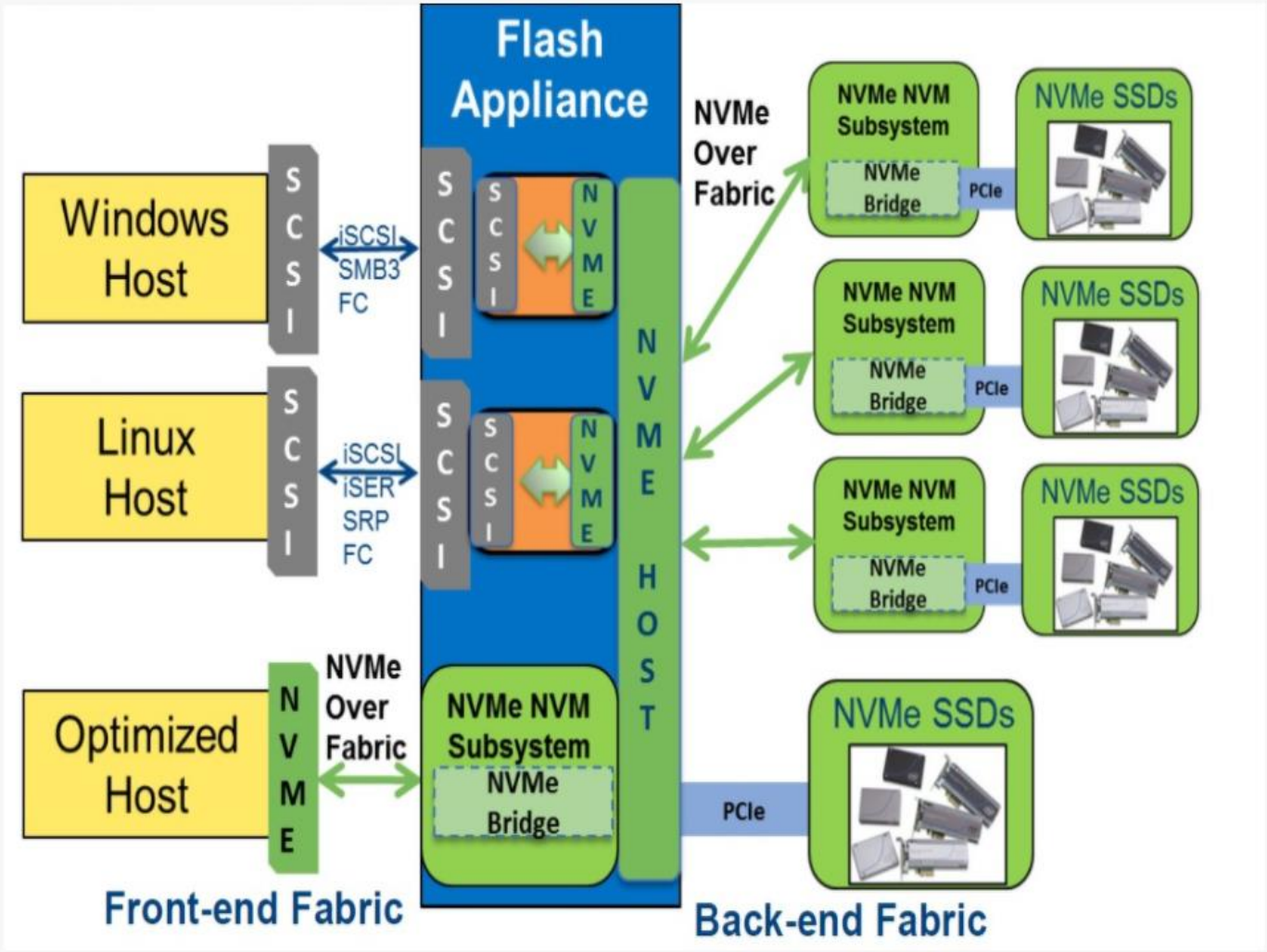


NEME Development

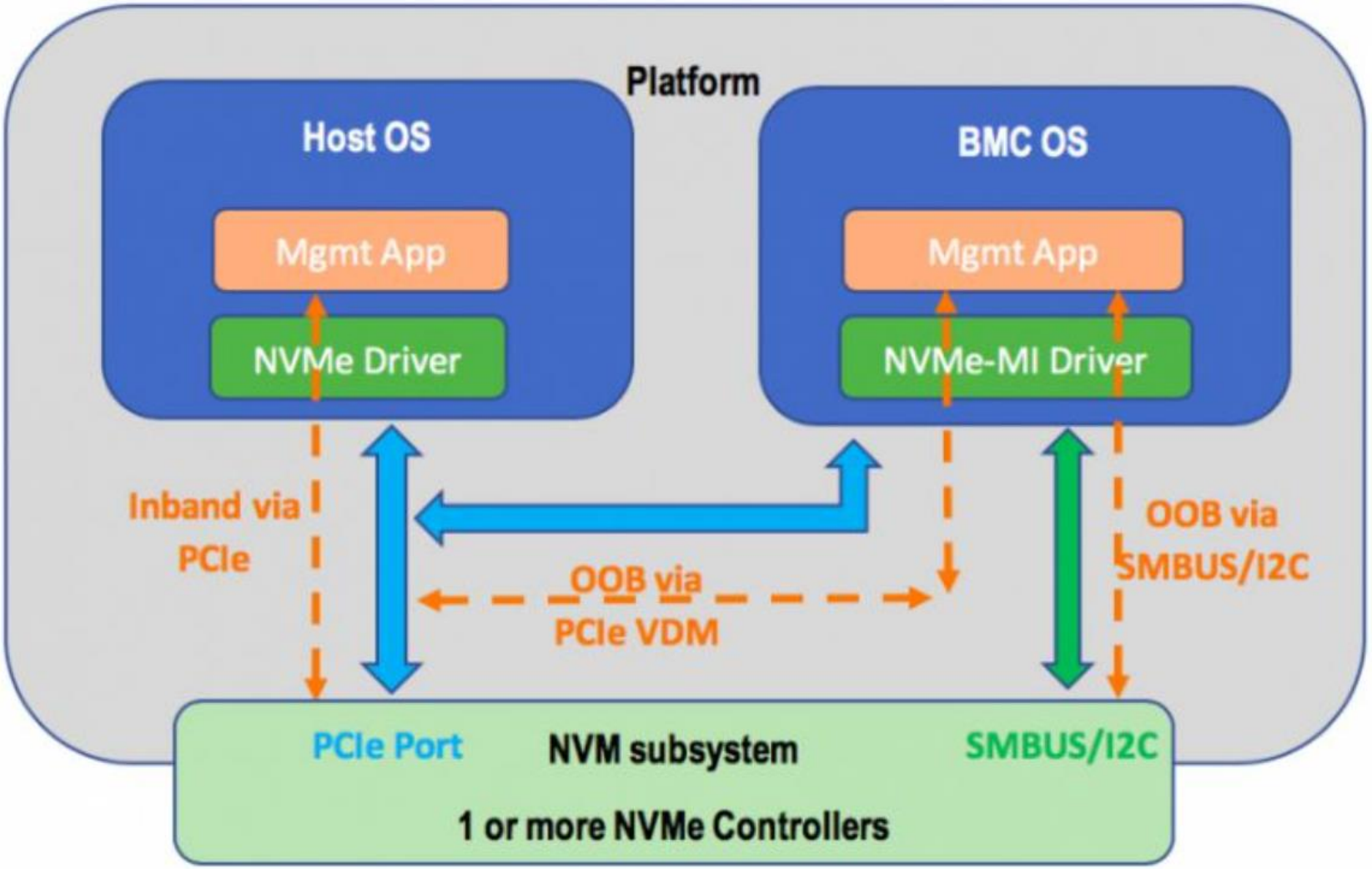
- NVME Spec
- NVME-OF Spec
- NVME-MI



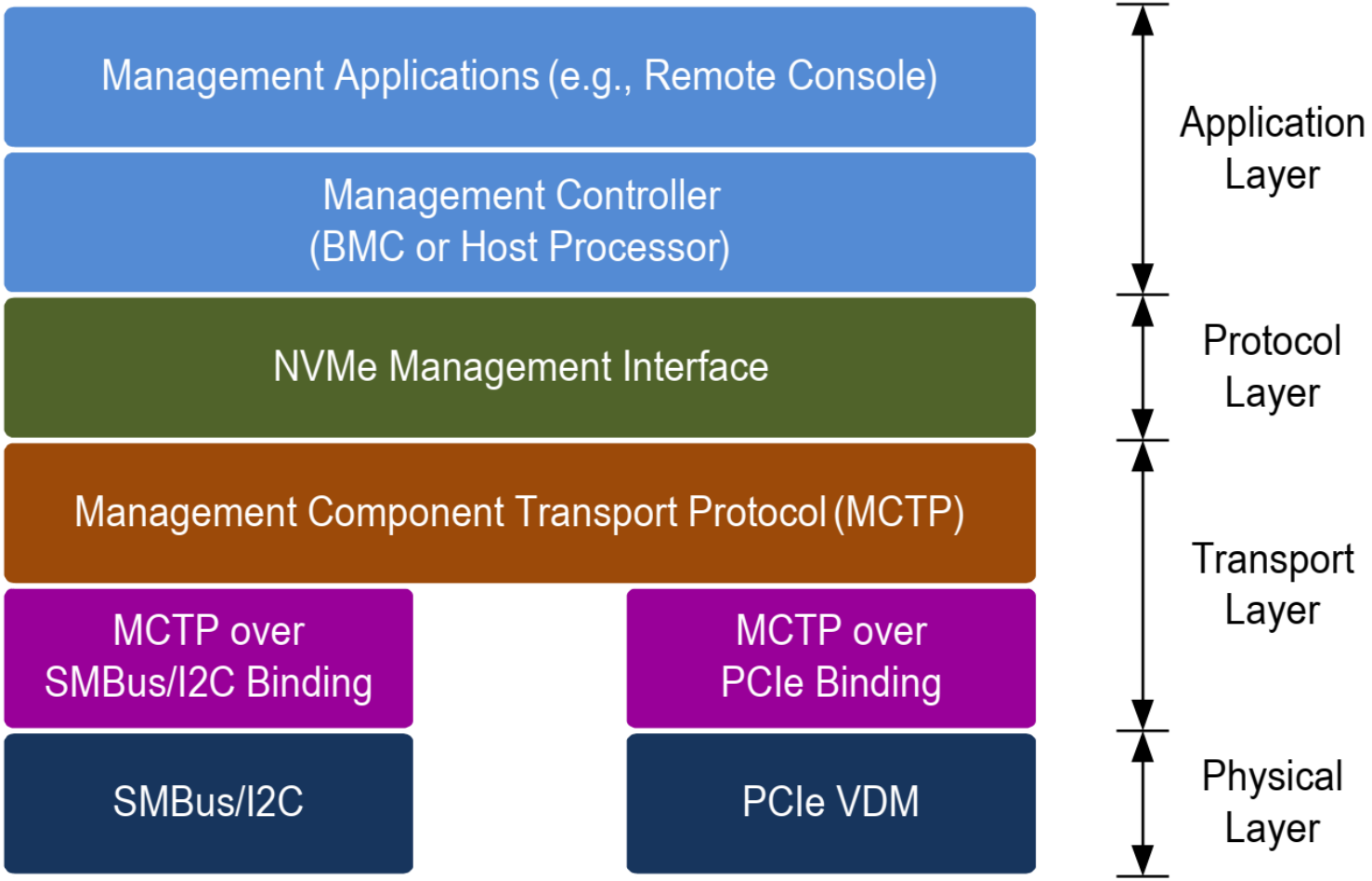
End-to-End NVMe over Fabric architecture



NVME-MI


Platform
Host OS
Mgmt App
NVMe Driver
BMC OS
Mgmt App
NVMe-MI Driver
Inband via
PCIe
OOB via
PCIe VDM
OOB via
SMBUS/I2C
PCIe Port
NVM subsystem
SMBUS/I2C
1 or more NVMe Controllers

NVME-MI



Introduction

- The NVM ExpressTM (NVMeTM) interface allows host software to communicate with a non-volatile memory subsystem. This interface is optimized for Enterprise and Client solid state drives, typically attached as a register level interface to the PCI Express interface.

Key attributes

- Does not require uncacheable / MMIO register reads in the command submission or completion path;
- A maximum of one MMIO register write is necessary in the command submission path;
- Support for up to 65,535 I/O Queues, with each I/O Queue supporting up to 65,535 outstanding commands;
- All information to complete a 4 KiB read request is included in the 64B command itself, ensuring efficient small I/O operation;
- Efficient and streamlined command set;
- Support for MSI/MSI-X and interrupt aggregation;
- Support for multiple namespaces;
- Efficient support for I/O virtualization architectures like SR-IOV;
- Robust error reporting and management capabilities; and
- Support for multi-path I/O and namespace sharing.

Submission & Completion Queue

Figure 1: Queue Pair Example, 1:1 Mapping

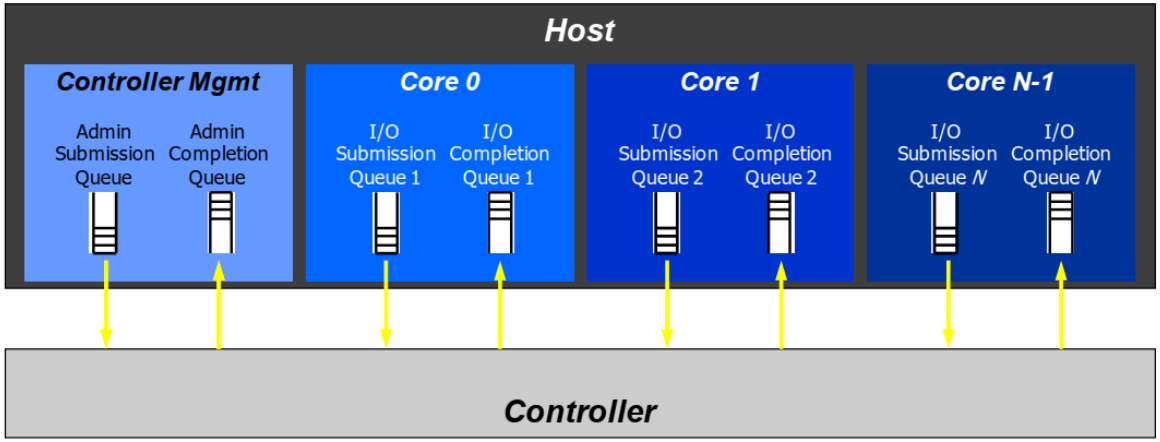
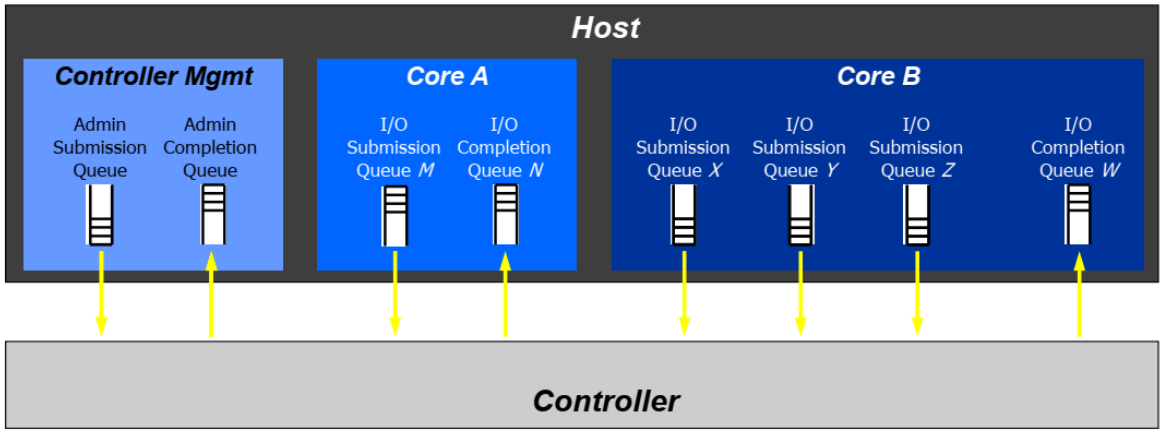


Figure 2: Queue Pair Example, *n*:1 Mapping



Multi-Path I/O and Namespace Sharing

Figure 3: NVM Express Controller with Two Namespaces

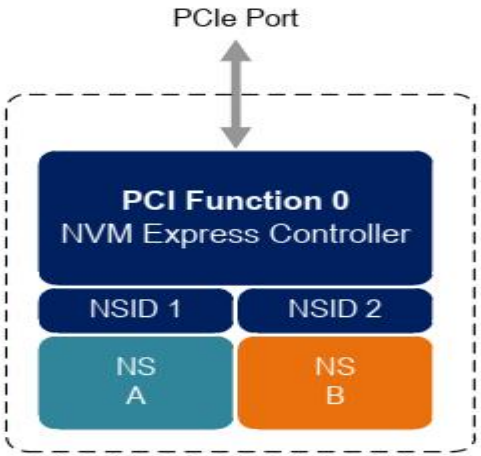
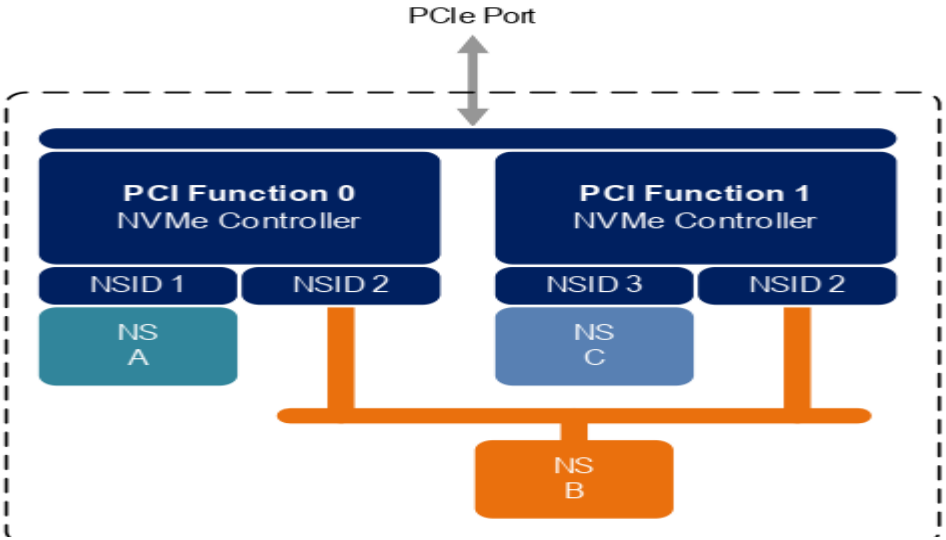
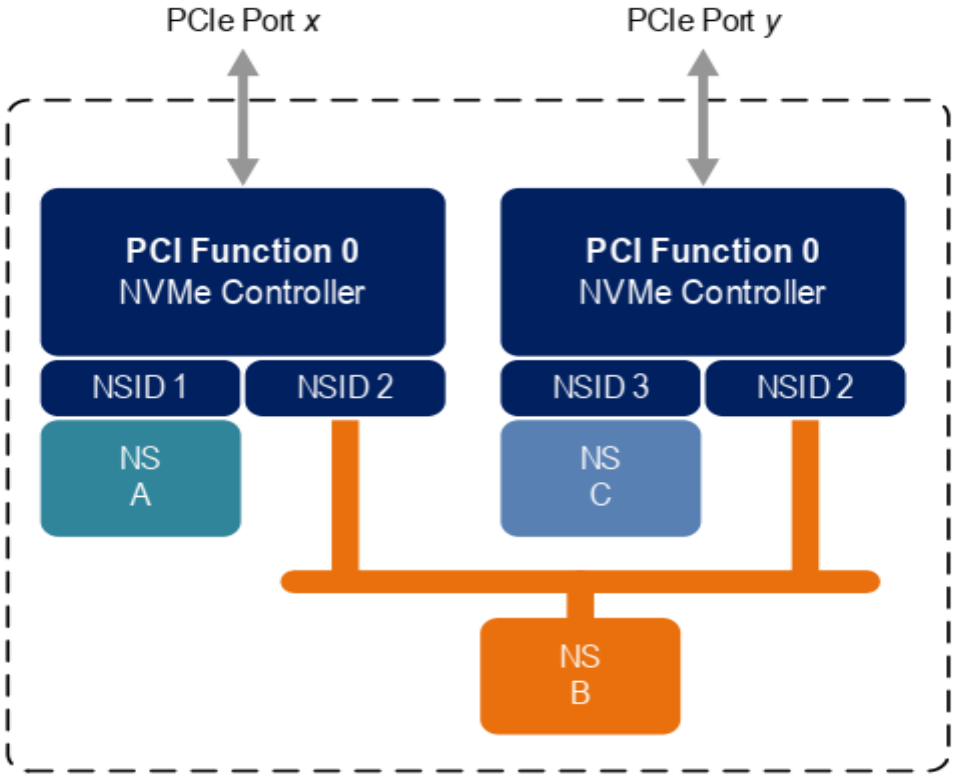


Figure 4: NVM Subsystem with Two Controllers and One Port



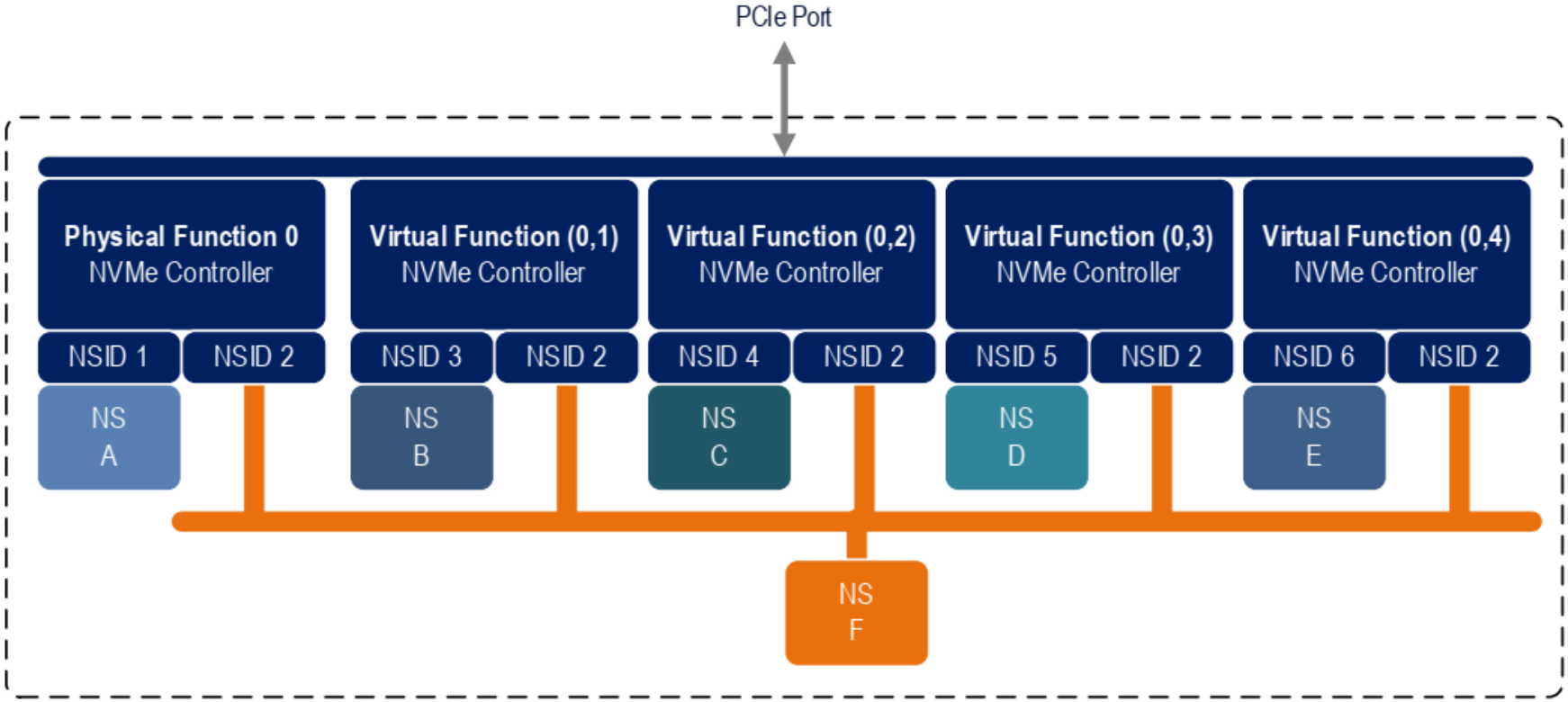
Multi-Path I/O and Namespace Sharing

Figure 5: NVM Subsystem with Two Controllers and Two Ports

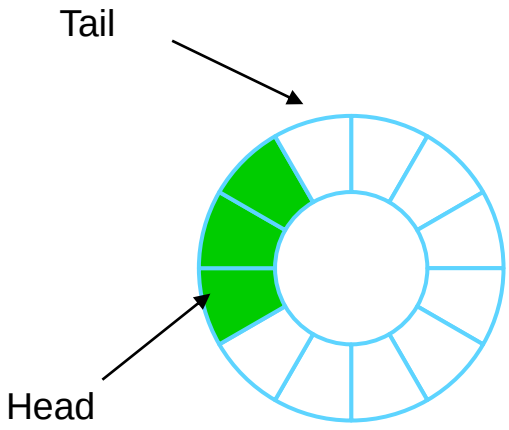


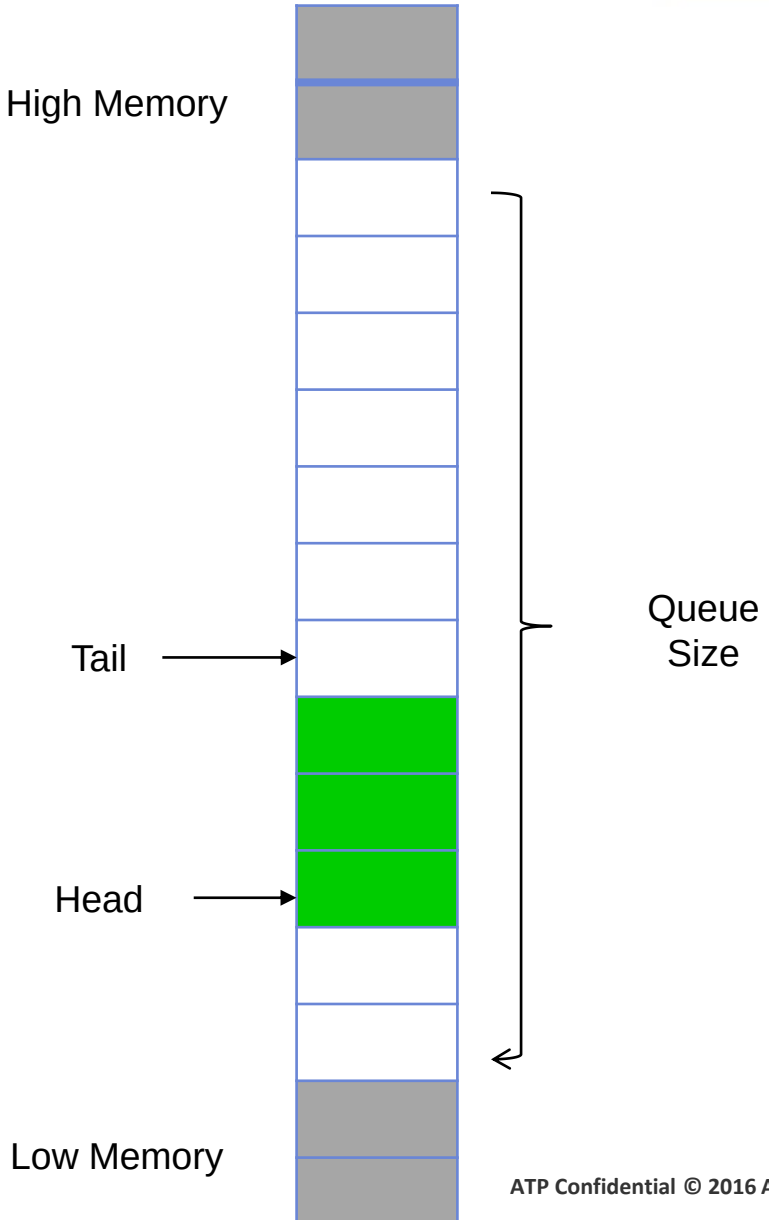
SR-IOV

Figure 6: PCI Express Device Supporting Single Root I/O Virtualization (SR-IOV)



Queue Structure





Queue Size

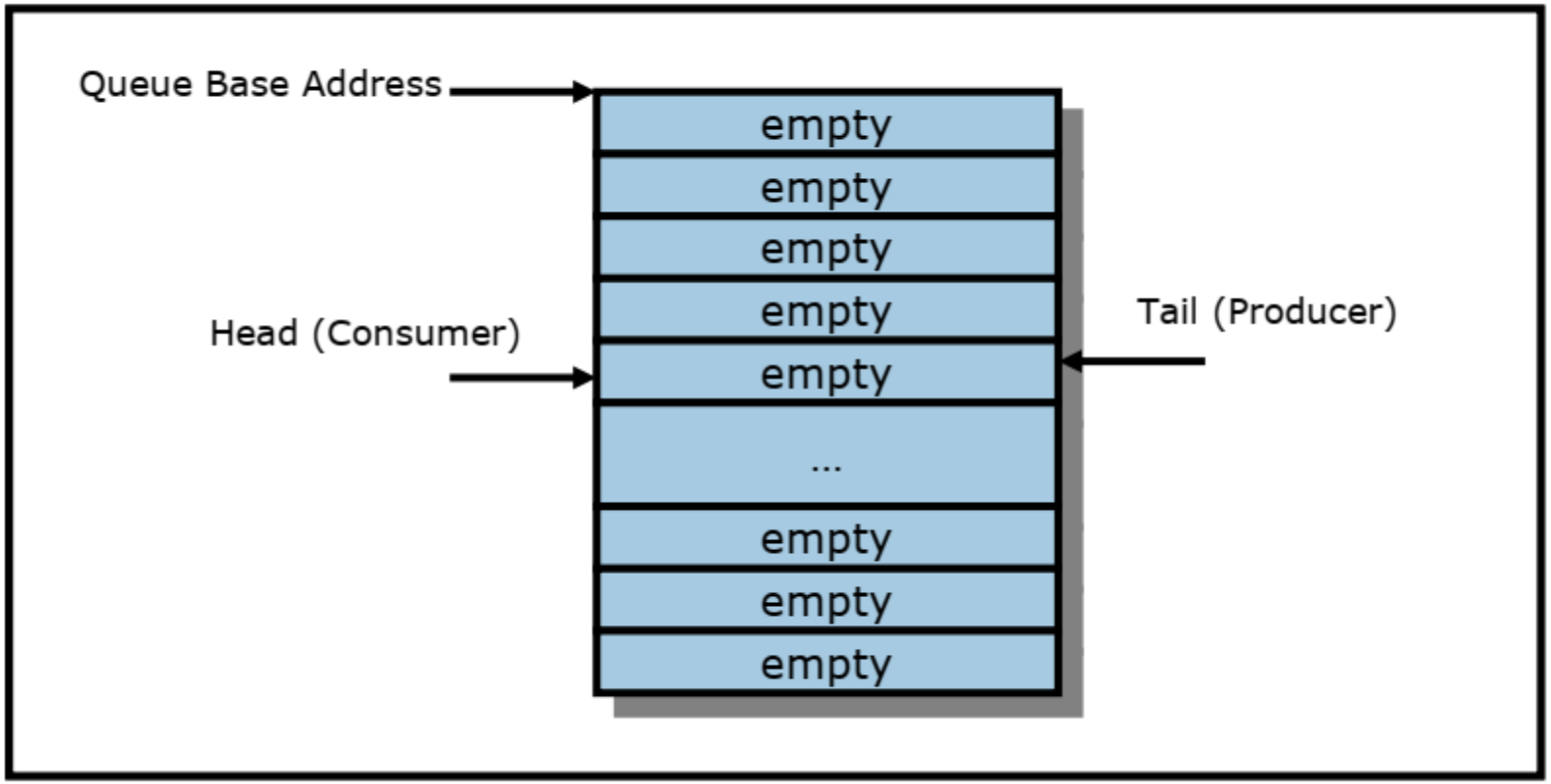
Figure 69: Offset 0h: CAP – Controller Capabilities

Bits	Type	Reset	Description
15:00	RO	Impl Spec	**Maximum Queue Entries Supported (MQES):** This field indicates the maximum individual queue size that the controller supports. For NVMe over PCIe implementations, this value applies to the I/O Submission Queues and I/O Completion Queues that the host creates. For NVMe over Fabrics implementations, this value applies to only the I/O Submission Queues that the host creates. This is a 0's based value. The minimum value is 1h, indicating two entries.

- **2 ^ 16 = 65535 = 64K**
- The maximum size for the Admin Submission and Admin Completion Queue **= 4K**
- The maximum size for either an I/O Submission Queue or an I/O Completion Queue = 64K
- The minimum size for a queue is two slots

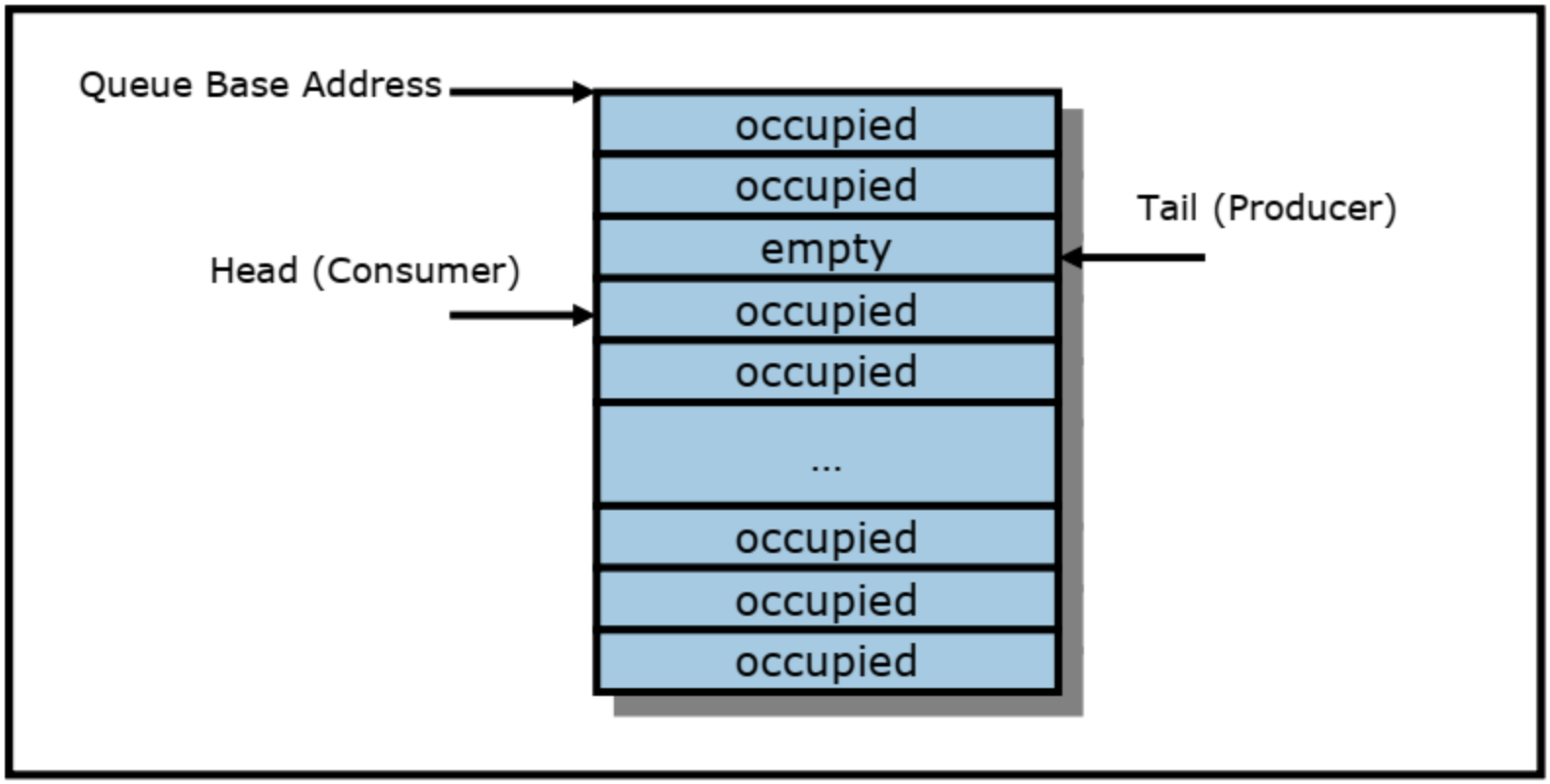
Empty Queue

Figure 102: Empty Queue Definition



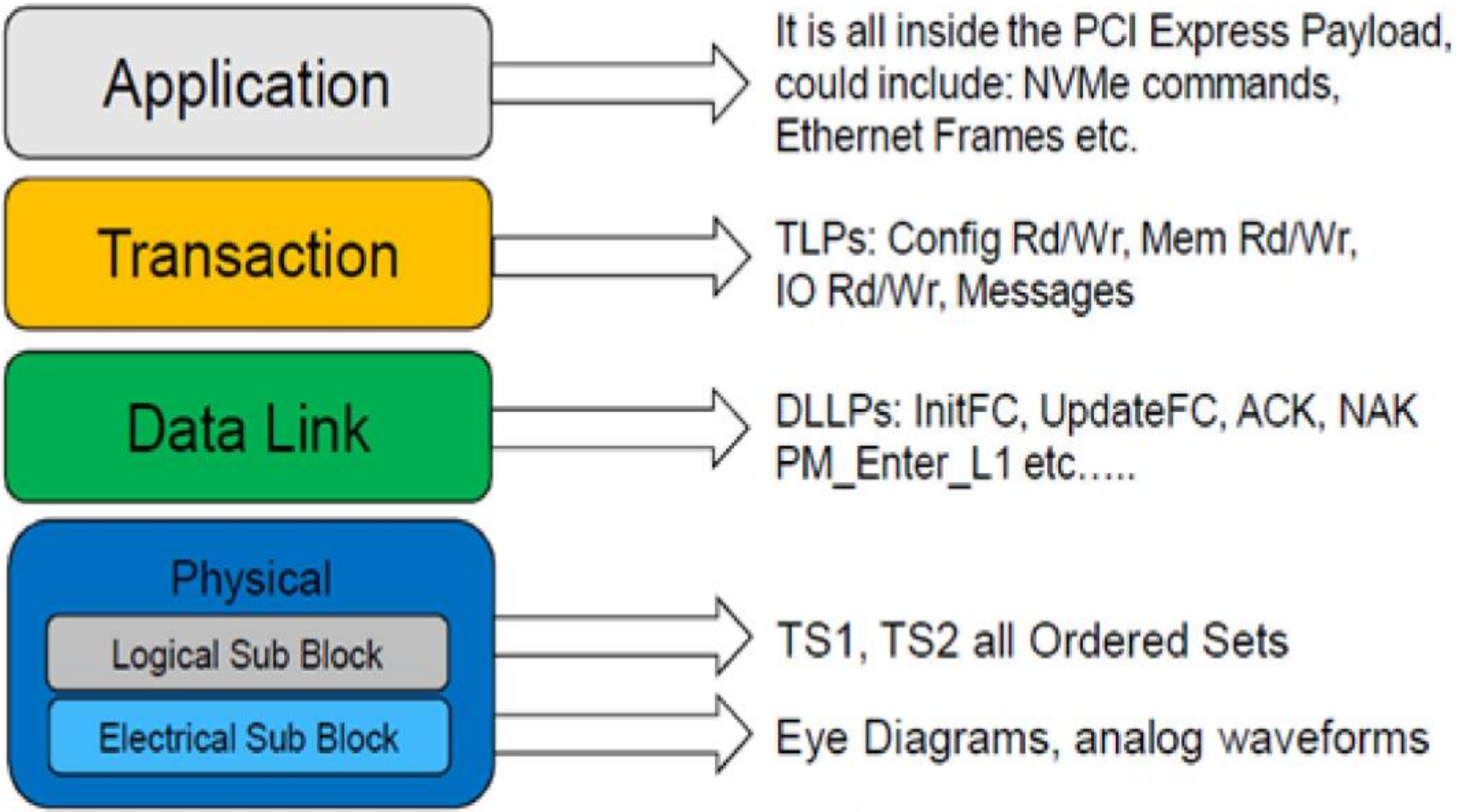
Full Queue

Figure 103: Full Queue Definition



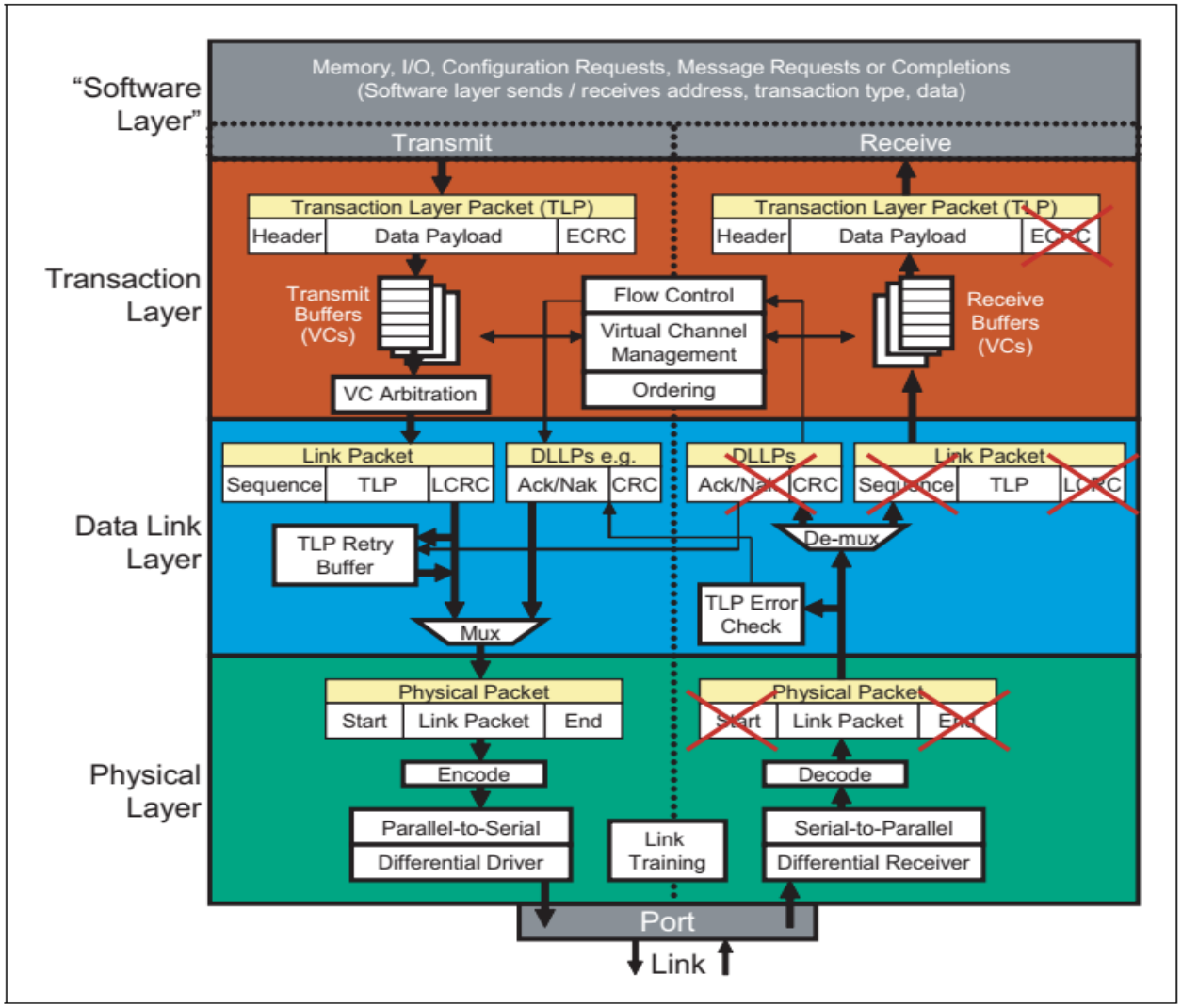
1.Queue number = Queue Size-1
2.Head = Tail +1

NVME & PCIE

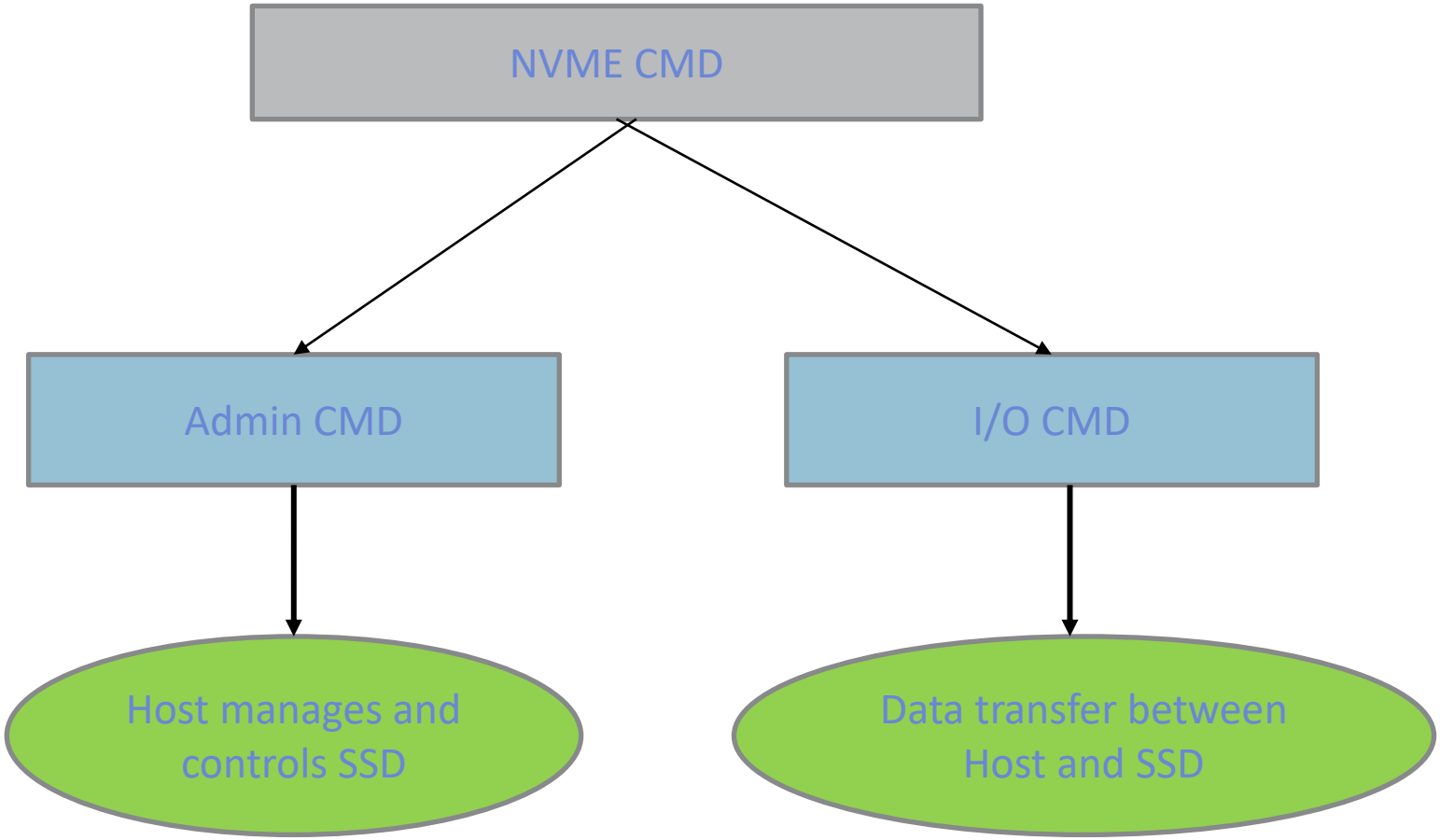


NVME & PCIE

Figure 2-14: Detailed Block Diagram of PCI Express Device's Layers



NVME Command



Admin CMD

Figure 139: Opcodes for Admin Commands

Opcode by Field			Combined Opcode [1]	Namespace Identifier Used [2]	Command
(07) Generic Command	(06:02) Function	(01:00) Data Transfer [3]			
0b	000 00b	00b	00h	No	Delete I/O Submission Queue
0b	000 00b	01b	01h	No	Create I/O Submission Queue
0b	000 00b	10b	02h	Yes	Get Log Page
0b	000 01b	00b	04h	No	Delete I/O Completion Queue
0b	000 01b	01b	05h	No	Create I/O Completion Queue
0b	000 01b	10b	06h	NOTE 6	Identify
0b	000 10b	00b	08h	No	Abort
0b	000 10b	01b	09h	Yes	Set Features
0b	000 10b	10b	0Ah	Yes	Get Features
0b	000 11b	00b	0Ch	No	Asynchronous Event Request
0b	000 11b	01b	0Dh	Yes	Namespace Management
0b	001 00b	00b	10h	No	Firmware Commit
0b	001 00b	01b	11h	No	Firmware Image Download
0b	001 01b	00b	14h	Yes	Device Self-test
0b	001 01b	01b	15h	Yes [4]	Namespace Attachment
0b	001 10b	00b	18h	No	Keep Alive
0b	001 10b	01b	19h	Yes [5]	Directive Send
0b	001 10b	10b	1Ah	Yes [5]	Directive Receive
0b	001 11b	00b	1Ch	No	Virtualization Management
0b	001 11b	01b	1Dh	No	NVMe-MI Send
0b	001 11b	10b	1Eh	No	NVMe-MI Receive
0b	111 11b	00b	7Ch	No	Doorbell Buffer Config
0b	111 11b	11b	7Fh	Refer to the NVMe over Fabrics specification.	
	I/O Command Set Specific				
1b	n/a	NOTE 3	80h to BFh		I/O Command Set specific
	Vendor Specific				
1b	n/a	NOTE 3	C0h to FFh		Vendor specific

NOTES:
1. Opcodes not listed are reserved.
2. A subset of commands use the Namespace Identifier (NSID) field. If the Namespace Identifier field is used, then the value FFFFFFFFh is supported in this field unless otherwise indicated in footnotes in this figure that a specific command does not support that value or supports that value only under specific conditions. When this field is not used, the field is cleared to 0h as described in Figure 105.
3. Indicates the data transfer direction of the command. All options to the command shall transfer data as specified or transfer no data. All commands, including vendor specific commands, shall follow this convention: 00b = no data transfer; 01b = host to controller; 10b = controller to host; 11b = bidirectional.
4. This command does not support the use of the Namespace Identifier (NSID) field set to FFFFFFFFh.
5. Support for the Namespace Identifier field set to FFFFFFFFh depends on the Directive Operation (refer to section 9).
6. Use of the Namespace Identifier field depends on the CNS value in the Identify Command as described in Figure 244.

Admin CMD

Figure 140: Opcodes for Admin Commands – NVM Command Set Specific

Opcode (07)	Opcode (06:02)	Opcode (01:00)	Opcode [1]	Namespace Identifier Used [2]	Command
Generic Command	Function	Data Transfer [3]			
1b	000 00b	00b	80h	Yes	Format NVM
1b	000 00b	01b	81h	NOTE 4	Security Send
1b	000 00b	10b	82h	NOTE 4	Security Receive
1b	000 01b	00b	84h	No	Sanitize
1b	000 01b	10b	86h	NOTE 5	Get LBA Status

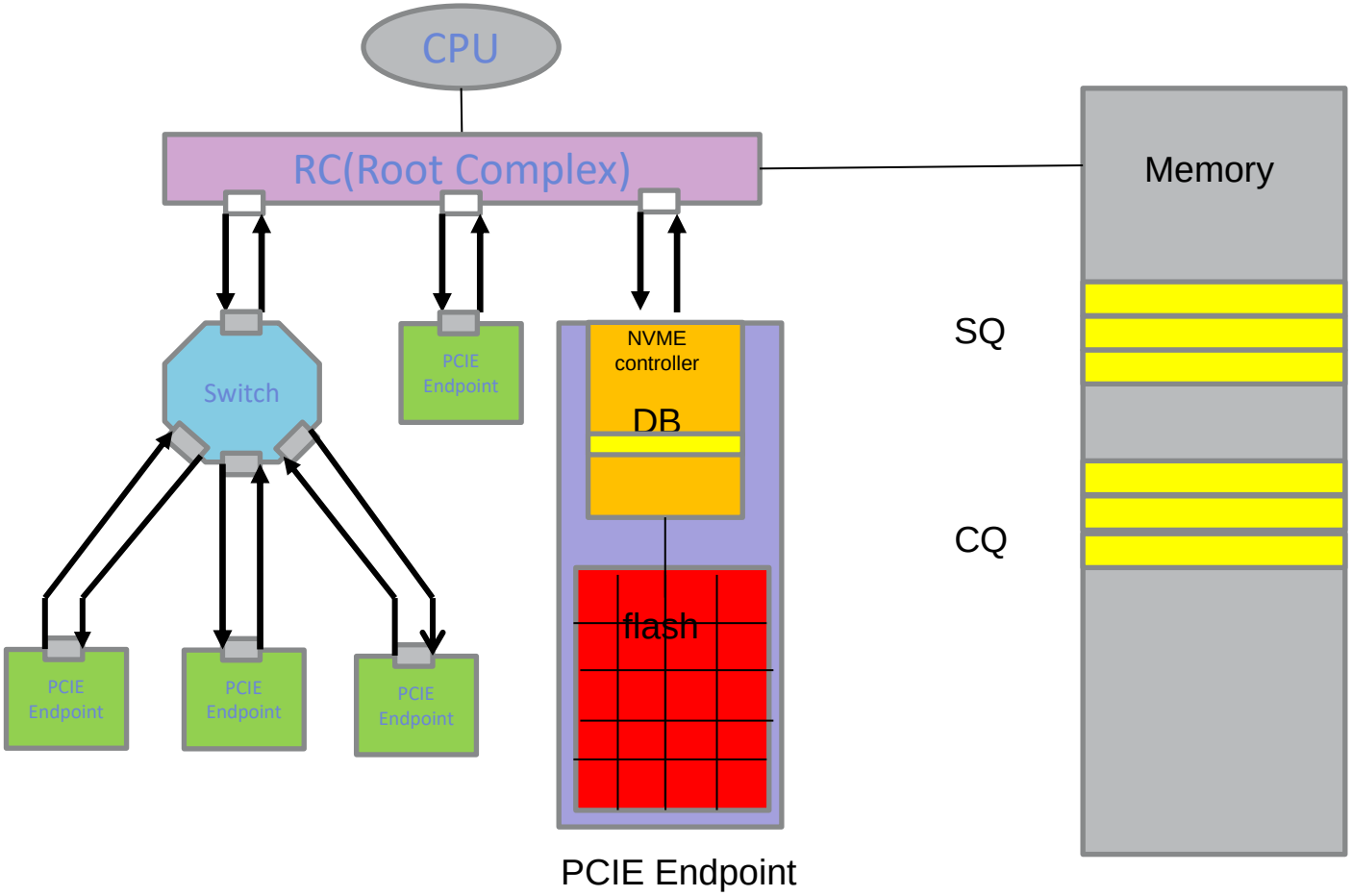
NOTES:
1. NVM Command Set Specific opcodes not listed are reserved.
2. A subset of commands use the Namespace Identifier (NSID) field. If the Namespace Identifier field is used, then unless otherwise specified, the value FFFFFFFFh is supported in this field. When this field is not used, the field is cleared to 0h as described in Figure 105.
3. Indicates the data transfer direction of the command. All options to the command shall transfer data as specified or transfer no data. All commands, including vendor specific commands, shall follow this convention: 00b = no data transfer; 01b = host to controller; 10b = controller to host; 11b = bidirectional.
4. The use of the Namespace Identifier is Security Protocol specific.
5. This command does not support the use of the Namespace Identifier (NSID) field set to FFFFFFFFh.

I/O CMD

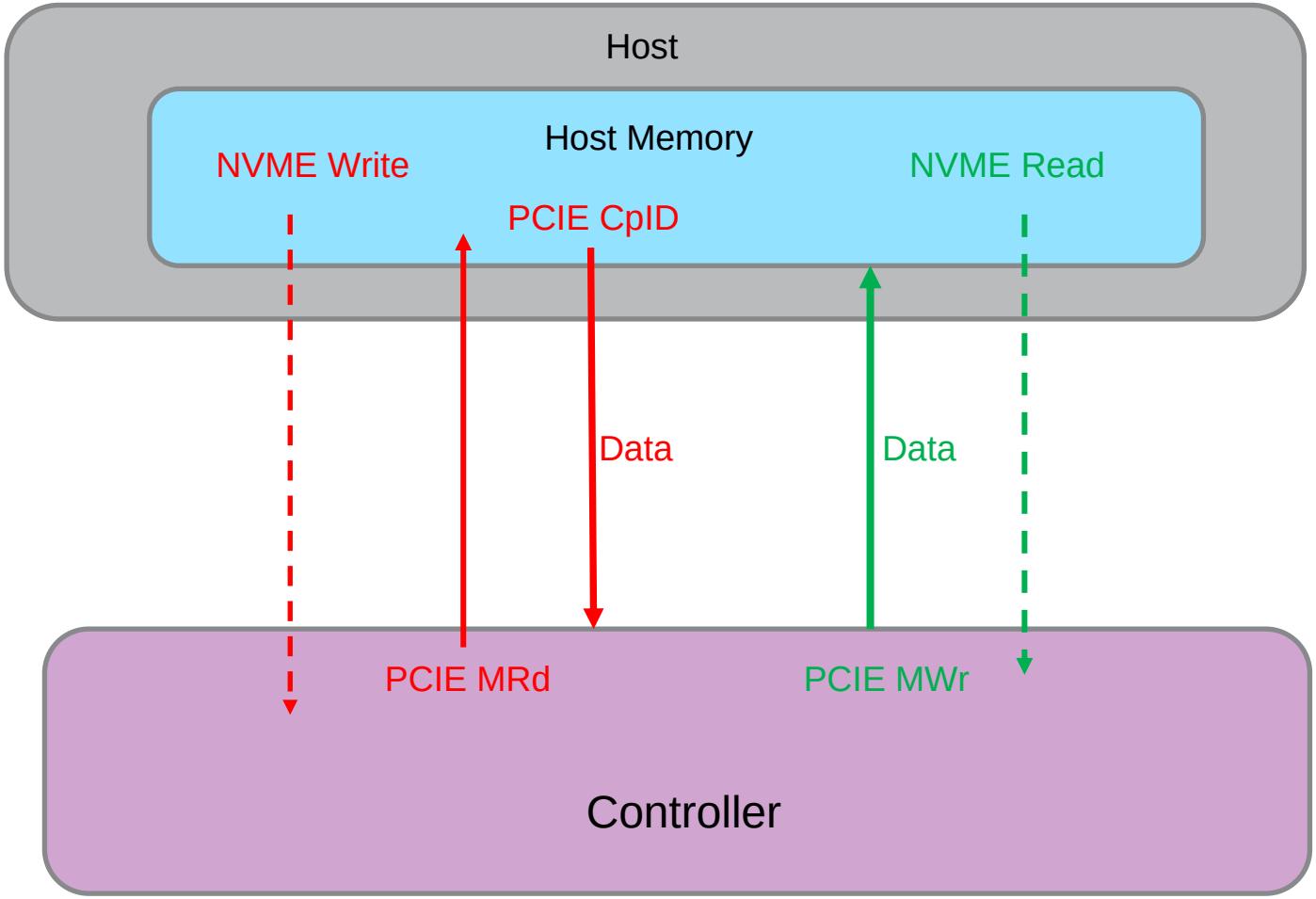
Figure 346: Opcodes for NVM Commands

Opcode by Field			Combined Opcode [1]	Command [2]	Reference Section
(07)	(06:02)	(01:00)			
Standard Command	Function	Data Transfer [3]			
0b	000 00b	00b	00h	Flush [4]	6.8
0b	000 00b	01b	01h	Write	6.15
0b	000 00b	10b	02h	Read	6.9
0b	000 01b	00b	04h	Write Uncorrectable	6.16
0b	000 01b	01b	05h	Compare	6.6
0b	000 10b	00b	08h	Write Zeroes	6.17
0b	000 10b	01b	09h	Dataset Management	6.7
0b	000 11b	00b	0Ch	Verify	6.14
0b	000 11b	01b	0Dh	Reservation Register	6.11
0b	000 11b	10b	0Eh	Reservation Report	6.13
0b	001 00b	01b	11h	Reservation Acquire	6.10
0b	001 01b	01b	15h	Reservation Release	6.12
Vendor Specific					
1b	n/a	NOTE 3	80h to FFh	Vendor specific	

NOTES:
1. Opcodes not listed are reserved.
2. All NVM commands use the Namespace Identifier (NSID) field. The value FFFFFFFFh is not supported in this field unless footnote 4 in this figure indicates that a specific command does support that value.
3. Indicates the data transfer direction of the command. All options to the command shall transfer data as specified or transfer no data. All commands, including vendor specific commands, shall follow this convention: 00b = no data transfer; 01b = host to controller; 10b = controller to host; 11b = bidirectional.
4. This command may support the use of the Namespace Identifier (NSID) field set to FFFFFFFFh.


CPU
RC(Root Complex)
Memory
Switch
PCIE Endpoint
NVME controller
DB
SQ
CQ
flash
PCIE Endpoint
PCIE Endpoint
PCIE Endpoint
PCIE Endpoint
PCIE Endpoint

Data interaction between host and controller

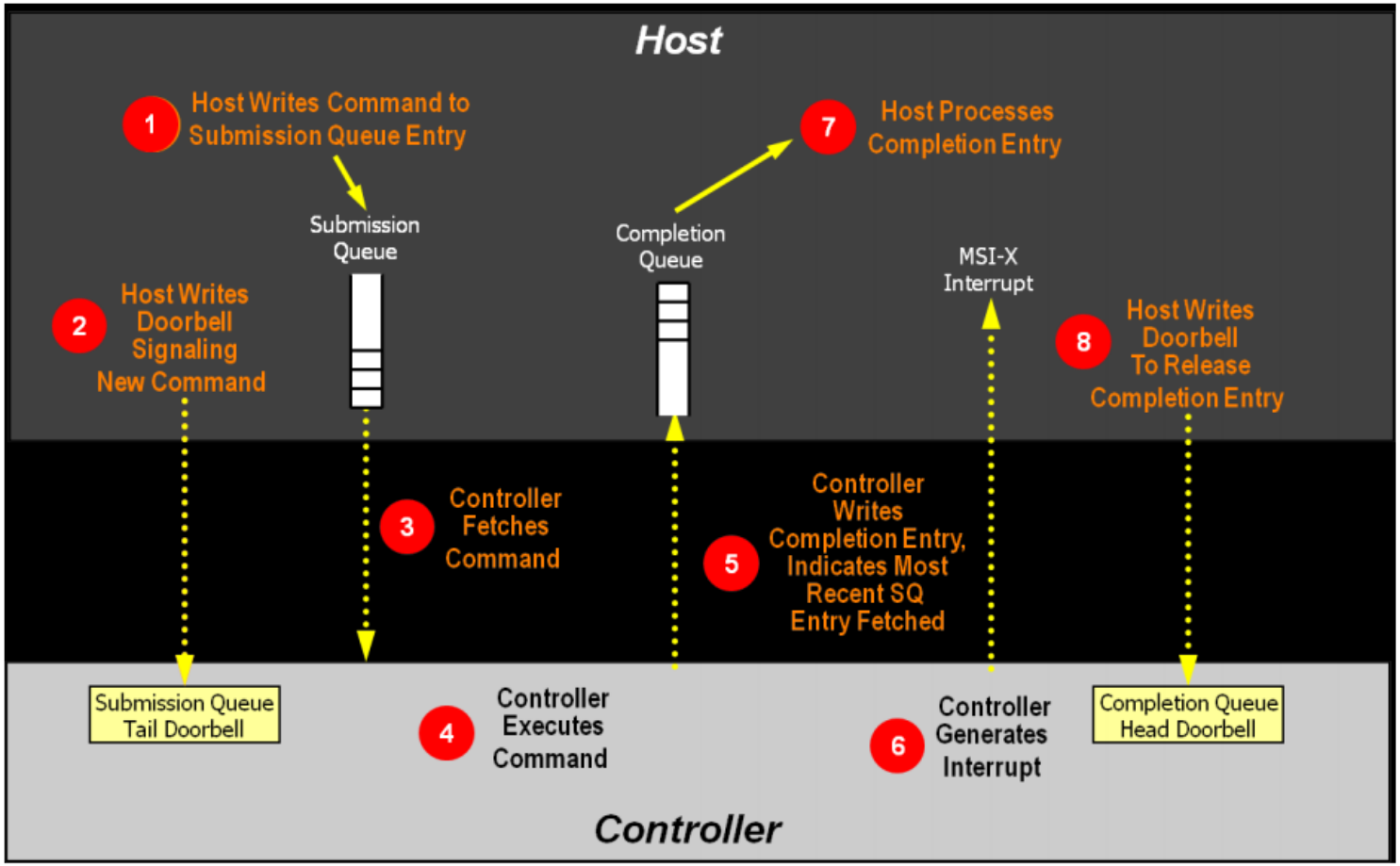


Command Processing

1. The host places one or more commands for execution in the next free Submission Queue slot(s) in memory;
2. The host updates the Submission Queue Tail Doorbell register with the new value of the Submission Queue Tail entry pointer. This indicates to the controller that a new command(s) is submitted for processing;
3. The controller transfers the command(s) from in the Submission Queue slot(s) into the controller for future execution. Arbitration is the method used to determine the Submission Queue from which the controller starts processing the next candidate command(s), refer to section 4.13;
4. The controller then proceeds with execution of the next command(s). Commands may complete out of order (the order submitted or started execution);
5. After a command has completed execution, the controller places a completion queue entry in the next free slot in the associated Completion Queue. As part of the completion queue entry, the controller indicates the most recent Submission Queue entry that has been consumed by advancing the Submission Queue Head pointer in the completion entry. Each new completion queue entry has a Phase Tag inverted from the previous entry to indicate to the host that this completion queue entry is a new entry;
6. The controller optionally generates an interrupt to the host to indicate that there is a new completion queue entry to consume and process. In the figure, this is shown as an MSI-X interrupt, however, it could also be a pin-based or MSI interrupt. Note that based on interrupt coalescing settings, an interrupt may or may not be generated for each new completion queue entry;
7. The host consumes and then processes the new completion queue entries in the Completion Queue. This includes taking any actions based on error conditions indicated. The host continues consuming and processing completion queue entries until a previously consumed entry with a Phase Tag inverted from the value of the current completion queue entries is encountered; and
8. The host writes the Completion Queue Head Doorbell register to indicate that the completion queue entry has been consumed. The host may consume many entries before updating the associated Completion Queue Head Doorbell register.

Command Processing

Figure 432: Command Processing



PRP & SGL

PRP:

A physical region page (PRP) entry is a pointer to a physical memory page. PRPs are used as a scatter/gather mechanism for data transfers between the controller and memory. To enable efficient out of order data transfers between the controller and the host, PRP entries are a fixed size.

SGL:

A Scatter Gather List (SGL) is a data structure in memory address space used to describe a data buffer.The controller indicates the SGL types that the controller supports in the Identify Controller data structure. A data buffer is either a source buffer or a destination buffer. An SGL contains one or more SGL segments.The total length of the Data Block and Bit Bucket descriptors in an SGL shall be equal to or exceed the amount of data required by the number of logical blocks transferred.

PRP

10:07	RW	0h	**Memory Page Size (MPS):** This field indicates the host memory page size. The memory page size is (2 ^ (12 + MPS)). Thus, the minimum host memory page size is 4 KiB and the maximum host memory page size is 128 MiB. The value set by host software shall be a supported value as indicated by the CAP.MPSMAX and CAP.MPSMIN fields. This field describes the value used for PRP entry size. This field shall only be modified when EN is cleared to '0'.

Figure 107: PRP Entry Layout

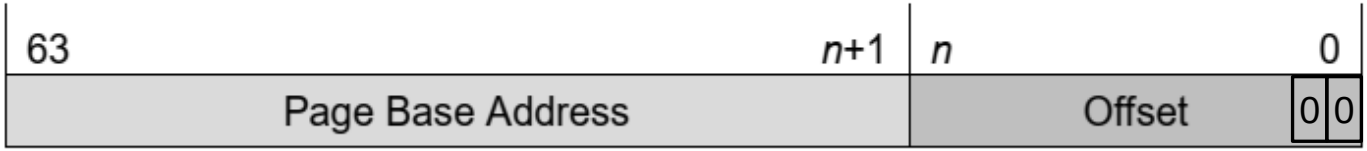
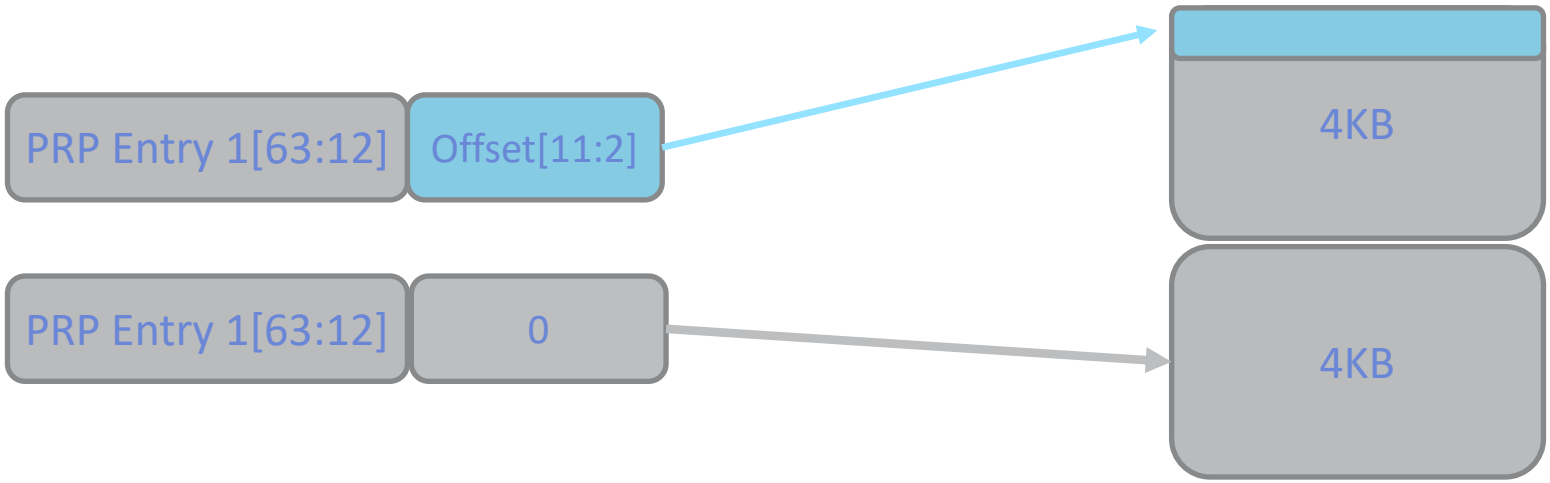


Figure 108: PRP Entry – Page Base Address and Offset

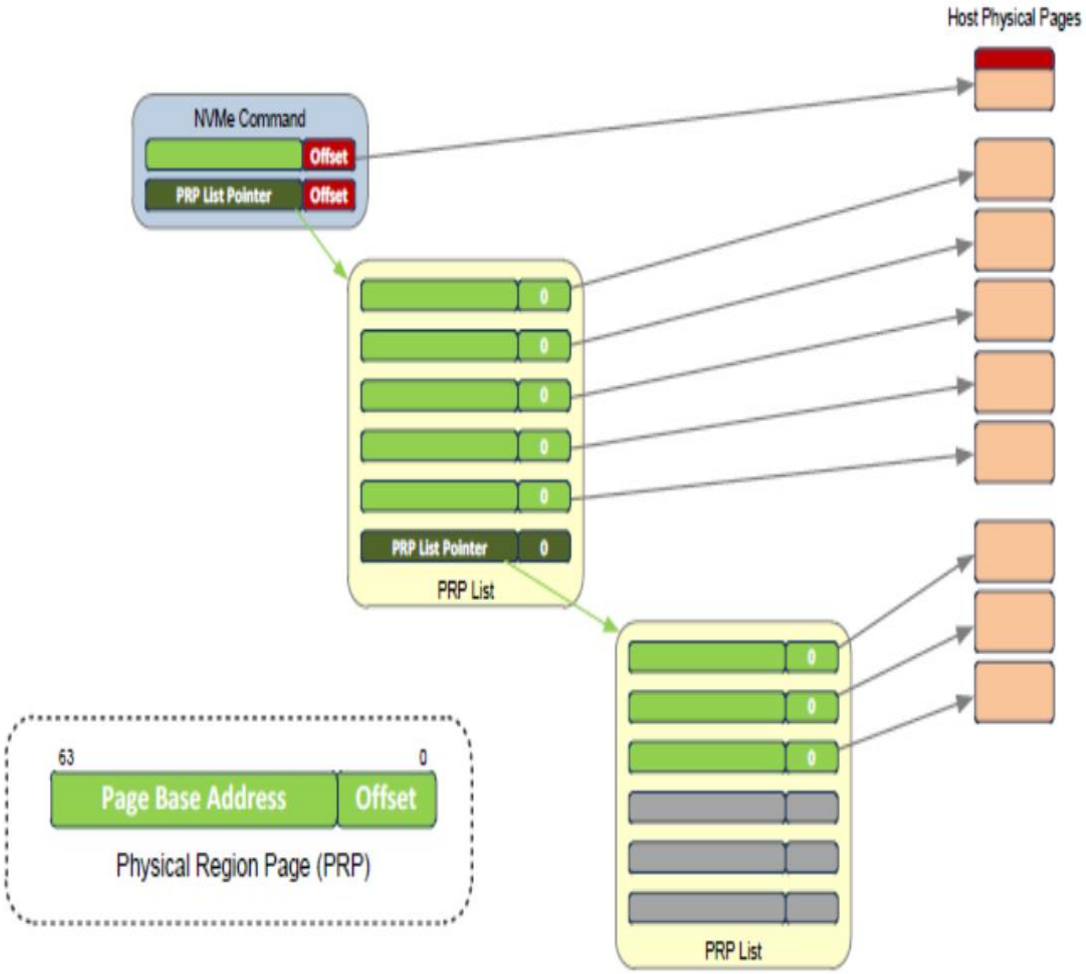
Bits	Description
63:00	**Page Base Address and Offset (PBAO):** This field indicates the 64-bit physical memory page address. The lower bits (*n*:0) of this field indicate the offset within the memory page (e.g., if the memory page size is 4 KiB, then bits 11:00 form the Offset; if the memory page size is 8 KiB, then bits 12:00 form the Offset). If this entry is not the first PRP entry in the command or a PRP List pointer in a command, then the Offset portion of this field shall be cleared to 0h. The Offset shall be dword aligned, indicated by bits 1:0 being cleared to 00b. Note: The controller is not required to check that bits 1:0 are cleared to 00b. The controller may report an error of PRP Offset Invalid if bits 1:0 are not cleared to 00b. If the controller does not report an error of PRP Offset Invalid, then the controller shall operate as if bits 1:0 are cleared to 00b.

PRP



- **If the physical page size is 4KB, a Command needs to transmit 12KB of data. At this time, two PRP Entry can point to a maximum of 8KB space, which does not meet the requirements. What should I do?**

PRP


Host Physical Pages
NVMe Command
Offset
PRP List Pointer
Offset
0
0
0
0
0
PRP List Pointer
0
PRP List
0
0
0
PRP List
63
0
Page Base Address
Offset
Physical Region Page (PRP)

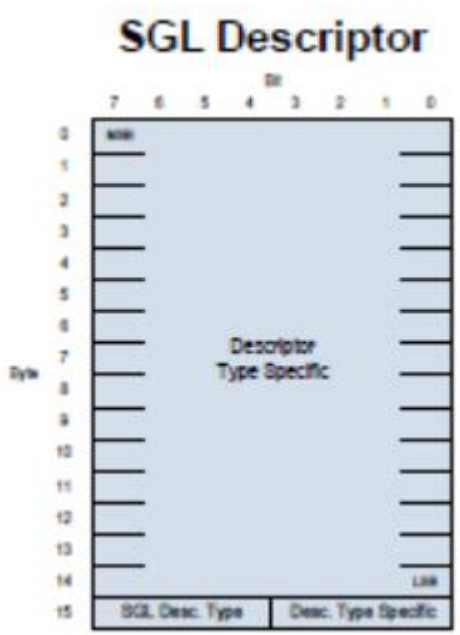
SGL

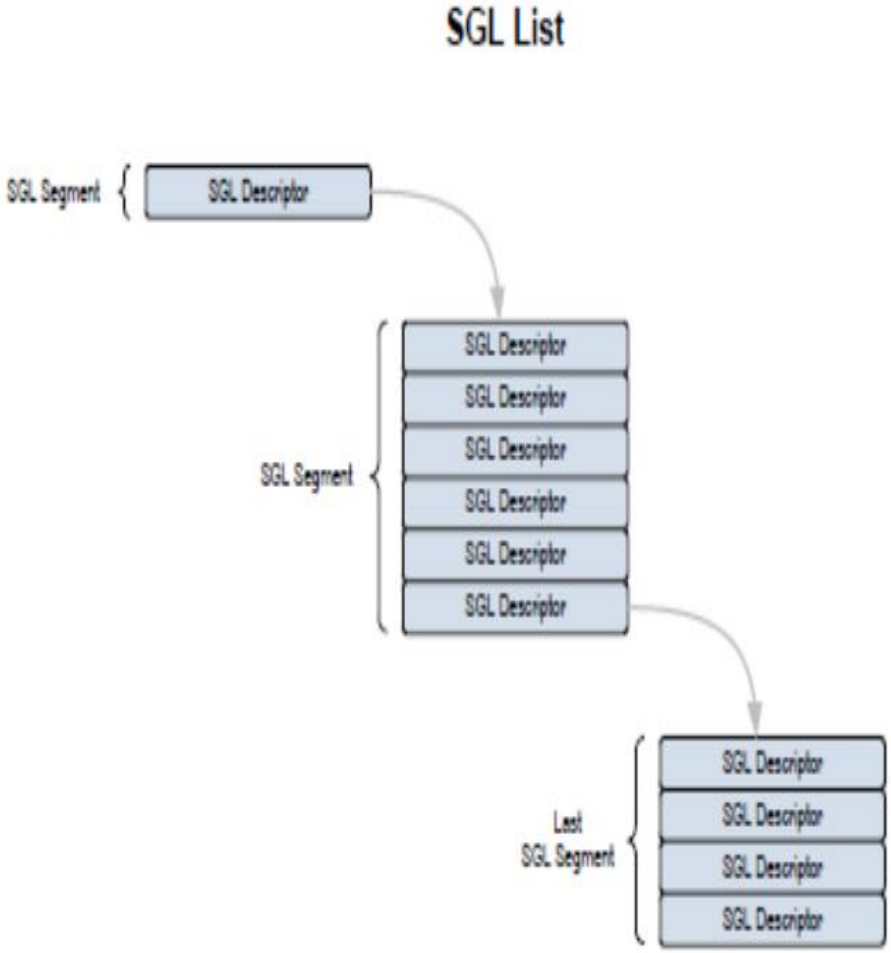
Bytes	Description
07:00	**Address:** If the SGL Identifier Descriptor Sub Type field is cleared to 0h, then the Address field specifies the starting 64-bit memory byte address of the next SGL segment, which is a SGL segment. If the SGL Identifier Descriptor Sub Type field is set to 1h, then the Address field contains an offset from the beginning of the location where data may be transferred.
11:08	**Length:** The Length field specifies the length in bytes of the next SGL segment. The Length field shall be a non-zero value and a multiple of 16. If the value in the Address field plus the value in the Length field is greater than 1_00000000_00000000h, then the SGL Segment descriptor shall be processed as having a Data SGL Length Invalid or Metadata SGL Length Invalid error.
14:12	Reserved
15	**SGL Identifier:** The definition of this field is described in the table below. **Bits** / **Description** 03:00 — **SGL Descriptor Sub Type:** Valid values are specified in Figure 113. 07:04 — **SGL Descriptor Type:** 2h as specified in Figure 112.

Figure 112: SGL Descriptor Type

Code	Descriptor
0h	SGL Data Block descriptor
1h	SGL Bit Bucket descriptor
2h	SGL Segment descriptor
3h	SGL Last Segment descriptor
4h	Keyed SGL Data Block descriptor
5h	Transport SGL Data Block descriptor
6h to Eh	Reserved
Fh	Vendor specific

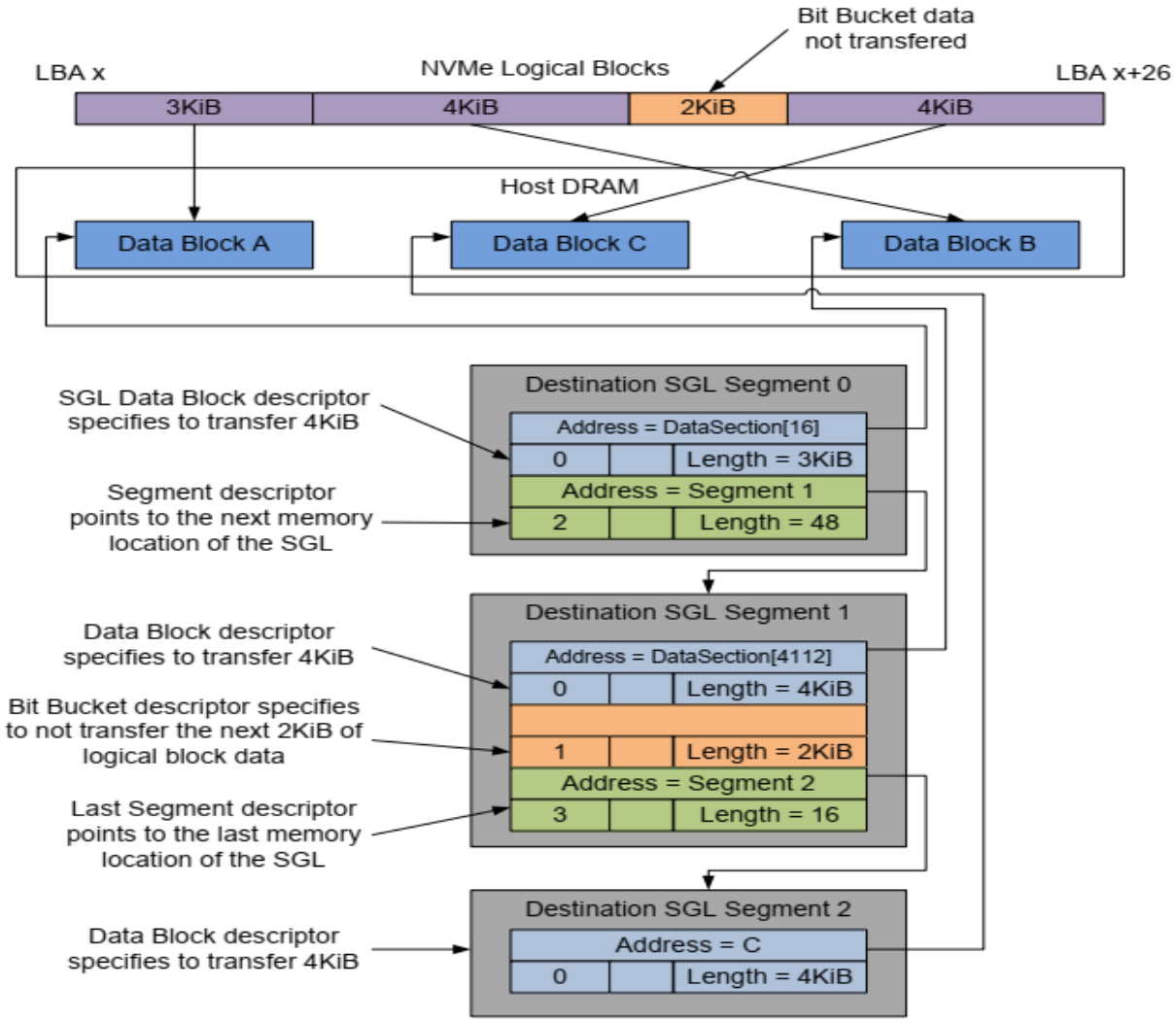
SGL


SGL Descriptor
Bit
7 6 5 4 3 2 1 0
Byte
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
Descriptor Type Specific
SGL Desc. Type
Desc. Type Specific

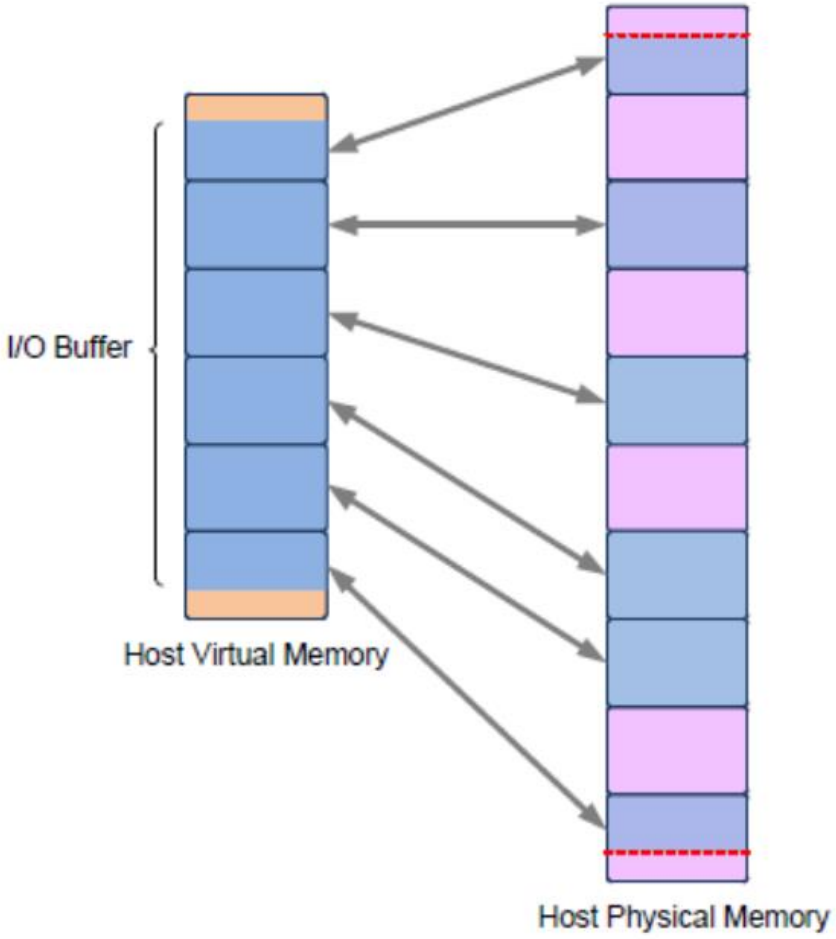

SGL List
SGL Segment
SGL Descriptor
SGL Segment
SGL Descriptor
SGL Descriptor
SGL Descriptor
SGL Descriptor
SGL Descriptor
SGL Descriptor
Last SGL Segment
SGL Descriptor
SGL Descriptor
SGL Descriptor
SGL Descriptor

SGL

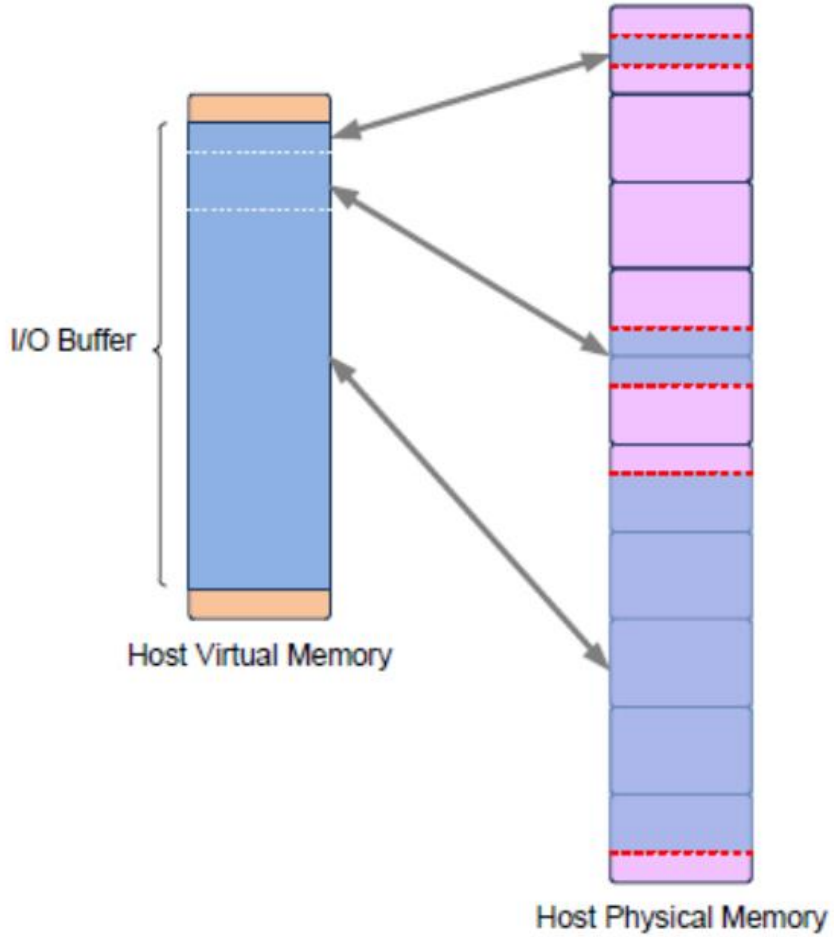
Figure 120: SGL Read Example



Compare PRD & SGL



PRP Data Transfer

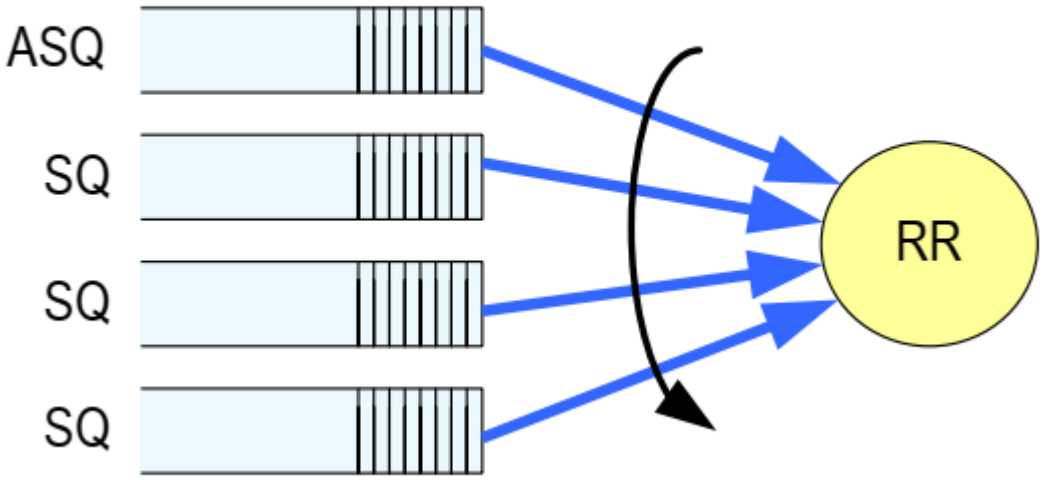


SGL Data Transfer

Command Arbitration

- **Round Robin Arbitration**

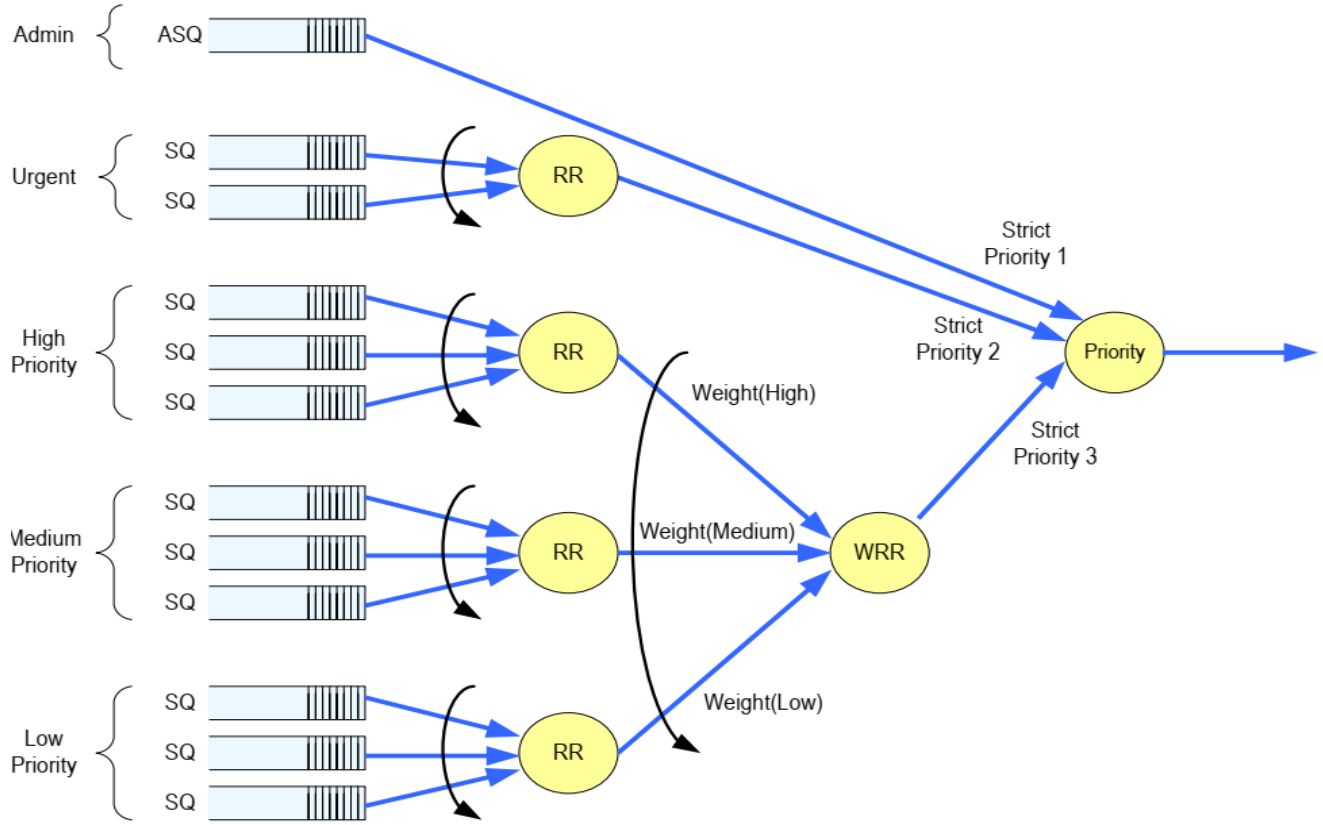
Figure 137: Round Robin Arbitration



Command Arbitration

- **Weighted Round Robin with Urgent Priority Class Arbitration**

Figure 138: Weighted Round Robin with Urgent Priority Class Arbitration



Command Arbitration

Figure 271: Set Features – Feature Identifiers

Feature Identifier	Current Setting Persists Across Power Cycle and Reset [2]	Uses Memory Buffer for Attributes	Feature Name
00h	Reserved		
01h	No	No	Arbitration
02h	No	No	Power Management
03h	Yes	Yes	LBA Range Type
04h	No	No	Temperature Threshold
05h	No	No	Error Recovery
06h	No	No	Volatile Write Cache
07h	No	No	Number of Queues
08h	No	No	Interrupt Coalescing
09h	No	No	Interrupt Vector Configuration
0Ah	No	No	Write Atomicity Normal
0Bh	No	No	Asynchronous Event Configuration
0Ch	No	Yes	Autonomous Power State Transition
0Dh	No [3]	No [4]	Host Memory Buffer
0Eh	No	Yes	Timestamp
0Fh	No	No	Keep Alive Timer
10h	Yes	No	Host Controlled Thermal Management
11h	No	No	Non-Operational Power State Config
12h	Yes	No	Read Recovery Level Config
13h	No	Yes	Predictable Latency Mode Config
14h	No	No	Predictable Latency Mode Window
15h	No	No	LBA Status Information Report Interval
16h	No	Yes	Host Behavior Support
17h	Yes	No	Sanitize Config
18h	No	No	Endurance Group Event Configuration
19h to 77h	Reserved		
78h to 7Fh	Refer to the NVMe Management Interface Specification for definition.		
80h to BFh			Command Set Specific (Reserved)

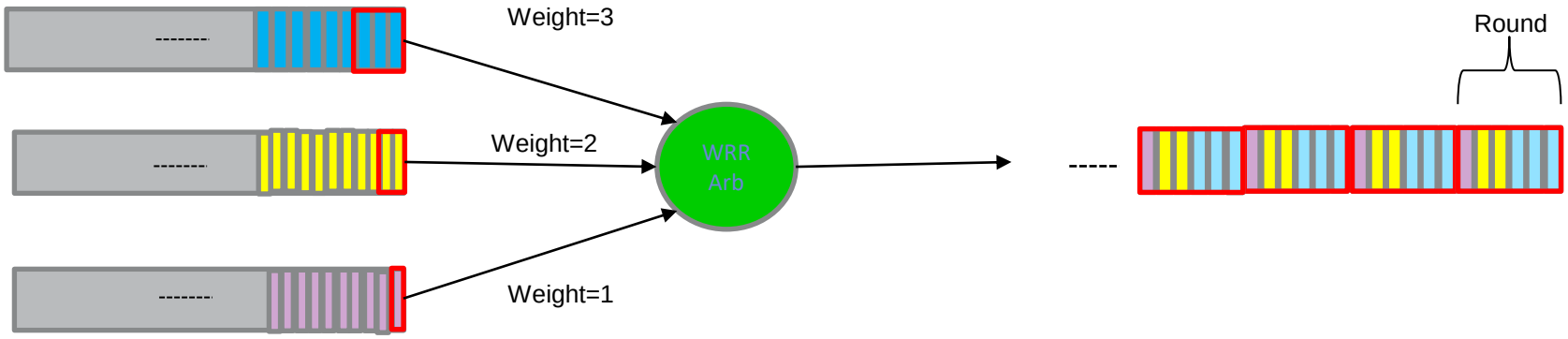
Command Arbitration

Figure 273: Arbitration & Command Processing – Command Dword 11

Bits	Description
31:24	**High Priority Weight (HPW):** This field defines the number of commands that may be executed from the high priority service class in each arbitration round. This is a 0's based value.
23:16	**Medium Priority Weight (MPW):** This field defines the number of commands that may be executed from the medium priority service class in each arbitration round. This is a 0's based value.
15:08	**Low Priority Weight (LPW):** This field defines the number of commands that may be executed from the low priority service class in each arbitration round. This is a 0's based value.
07:03	Reserved
02:00	**Arbitration Burst (AB):** Indicates the maximum number of commands that the controller may launch at one time from a particular Submission Queue. The value is expressed as a power of two (e.g., 000b indicates one, 011b indicates eight). A value of 111b indicates no limit.

Command Arbitration

- **EX:AB=0,HPW=3,MPW=2,LPW=1**



Interrupt

- The specification allows the controller to be configured to report interrupts in one of four modes.

1.pin-based interrupt.

2.single message MSI.

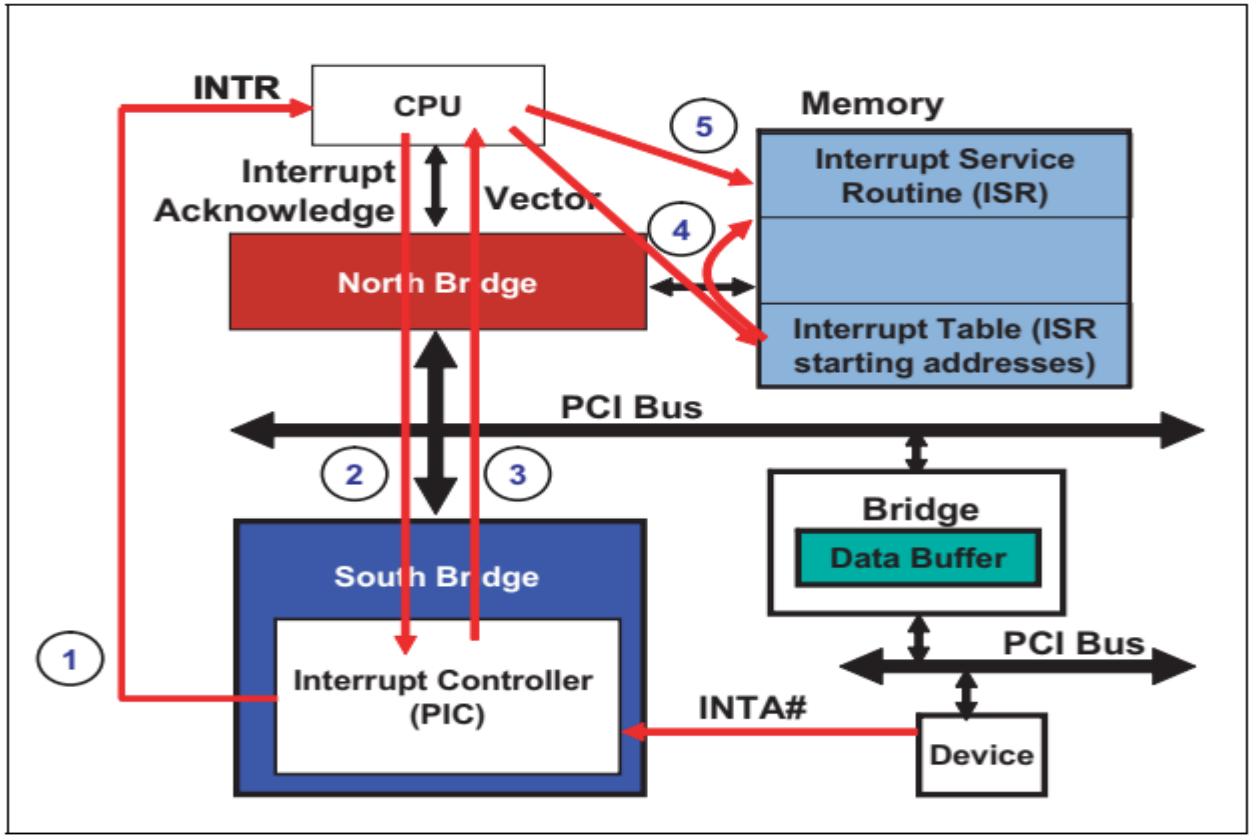
3.multiple message MSI.

4.MSI-X .

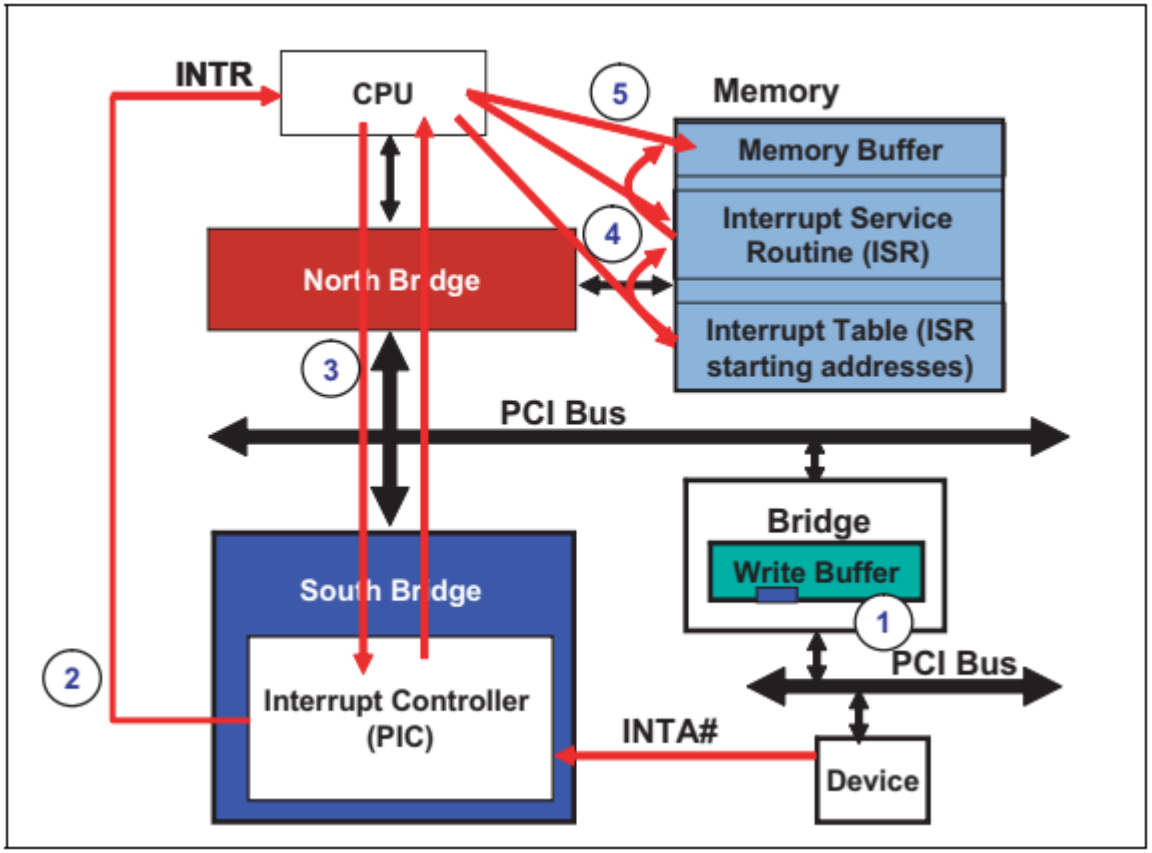
Pin-based interrupt

- MSICAP.MC.MSIE=0 ; MSICAP.MC.MME=000b.

Figure 17-3: Legacy Interrupt Example

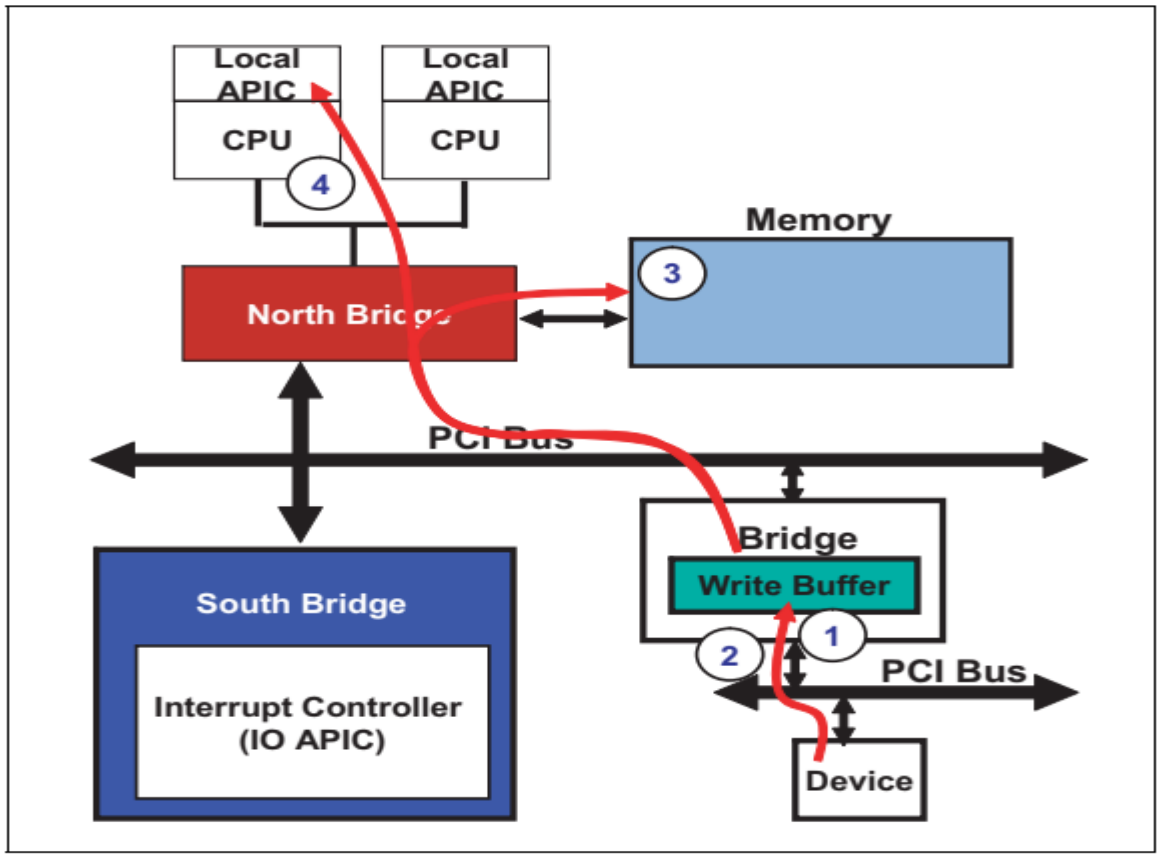


Memory Synchronization Problem



Memory Synchronization Problem

Figure 17-22: MSI Delivery



Single message MSI,Multiple message MSI& MSI-X

- Single message MSI
  MSICAP.MC.MSIE=1 ; MSICAP.MC.MME=000b.
- Multiple message MSI
  MSICAP.MC.MSIE=1; MSICAP.MC.MME=001b~101b
- MSI-X

(MSICAP.MC.MSIE='0') and (MSICAP.MC.MME=000b) and MSI-X is enabled

Figure 44: Offset MSIXCAP + 2h: MXC – MSI-X Message Control

Bits	Type	Reset	Description
15	RW	0b	**MSI-X Enable (MXE):** If set to '1' and the MSI Enable bit in the MSI Message Control register is cleared to '0', the function is permitted to use MSI-X to request service and is prohibited from using its INTx# pin (if implemented). If cleared to '0', the function is prohibited from using MSI-X to request service.
14	RW	0b	**Function Mask (FM):** If set to '1', all of the vectors associated with the function are masked, regardless of their per vector Mask bit states. If cleared to '0', each vector's Mask bit determines whether the vector is masked or not. Setting to '1' or clearing to '0' the MSI-X Function Mask bit has no effect on the state of the per vector Mask bits.
13:11	RO	000b	Reserved
10:00	RO	Impl Spec	**Table Size (TS):** This value indicates the size of the MSI-X Table as the value *n*, which is encoded as *n* - 1. For example, a returned value of 3h corresponds to a table size of 4.

Multiple MSI

- Multiple MSI may use up to 32 interrupt vectors.

Figure 36: Offset MSICAP + 2h: MC – Message Signaled Interrupt Message Control

Bits	Type	Reset	Description
15:09	RO	0h	Reserved
08	RO	Impl Spec	**Per-Vector Masking Capable (PVM):** Specifies whether controller supports MSI per-vector masking.
07	RO	1b	**64 Bit Address Capable (C64):** Specifies whether the controller is capable of generating 64-bit messages. NVM Express controllers shall be 64-bit capable.
06:04	RW	000b	**Multiple Message Enable (MME):** Indicates the number of messages the controller should assert. Controllers that only support single message MSI may implement this field as read-only.
03:01	RO	Impl Spec	**Multiple Message Capable (MMC):** Indicates the number of messages the controller is requesting.
00	RW	0b	**MSI Enable (MSIE):** If set to '1', MSI is enabled. If cleared to '0', MSI operation is disabled.

MSI-X

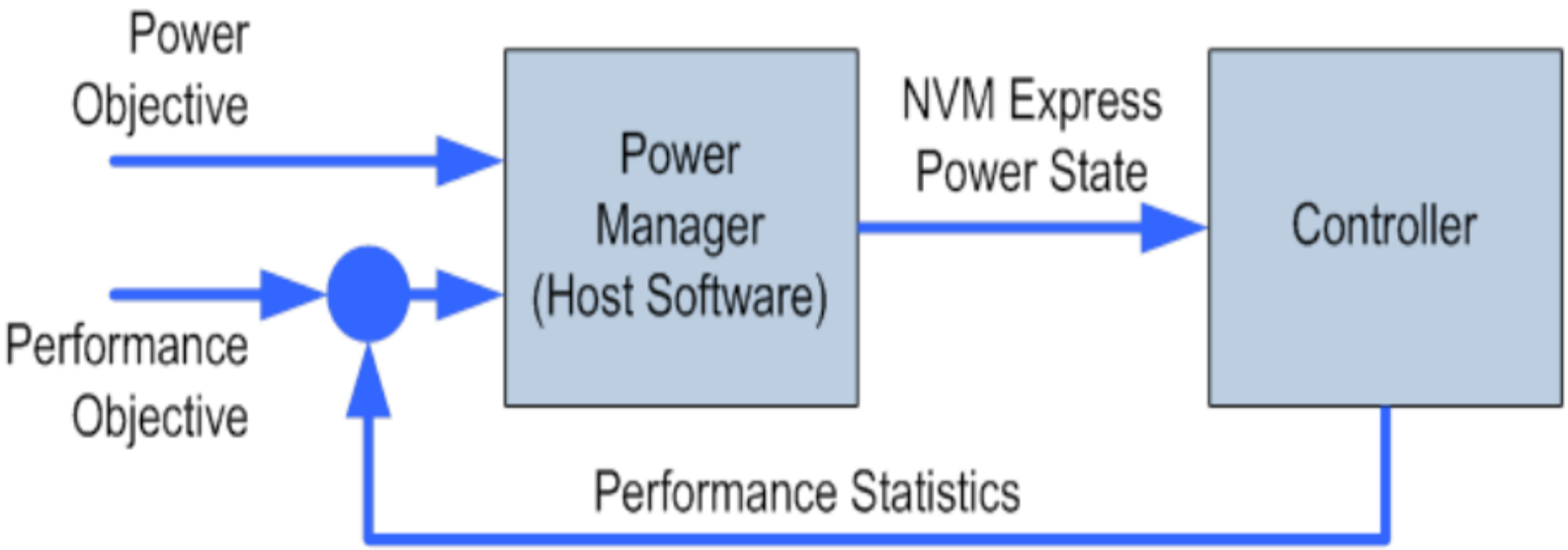
- The maximum number of vectors is 2K

Figure 44: Offset MSIXCAP + 2h: MXC – MSI-X Message Control

Bits	Type	Reset	Description
15	RW	0b	**MSI-X Enable (MXE):** If set to '1' and the MSI Enable bit in the MSI Message Control register is cleared to '0', the function is permitted to use MSI-X to request service and is prohibited from using its INTx# pin (if implemented). If cleared to '0', the function is prohibited from using MSI-X to request service.
14	RW	0b	**Function Mask (FM):** If set to '1', all of the vectors associated with the function are masked, regardless of their per vector Mask bit states. If cleared to '0', each vector's Mask bit determines whether the vector is masked or not. Setting to '1' or clearing to '0' the MSI-X Function Mask bit has no effect on the state of the per vector Mask bits.
13:11	RO	000b	Reserved
10:00	RO	Impl Spec	**Table Size (TS):** This value indicates the size of the MSI-X Table as the value *n*, which is encoded as *n* - 1. For example, a returned value of 3h corresponds to a table size of 4.

Power Management

Figure 455: Dynamic Power Management



Power Management

- The number of power states implemented by a controller is returned in the Number of Power States Supported (NPSS) field in the Identify Controller data structure. A controller shall support at least one power state and may optionally support up to a total of 32 power states.

263	M	**Number of Power States Support (NPSS):** This field indicates the number of NVM Express power states supported by the controller. This is a 0's based value. Refer to section 8.4. Power states are numbered sequentially starting at power state 0. A controller shall support at least one power state (i.e., power state 0) and may support up to 31 additional power states (i.e., up to 32 total).

32 Bytes Power State

Figure 248: Identify – Power State Descriptor Data Structure

<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>255:184</td><td>Reserved</td></tr>
<tr><td>183:182</td><td>Active Power Scale (APS): This field indicates the scale for the Active Power field. If an Active Power Workload is reported for a power state, then the Active Power Scale shall also be reported for that power state.
<table>
<tr><th>Value</th><th>Definition</th></tr>
<tr><td>00b</td><td>Not reported for this power state</td></tr>
<tr><td>01b</td><td>0.0001 W</td></tr>
<tr><td>10b</td><td>0.01 W</td></tr>
<tr><td>11b</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>181:179</td><td>Reserved</td></tr>
<tr><td>178:176</td><td>Active Power Workload (APW): This field indicates the workload used to calculate maximum power for this power state. Refer to section 8.4.3 for more details on each of the defined workloads. This field shall not be “No Workload” unless ACTP is 0h.</td></tr>
<tr><td>175:160</td><td>Active Power (ACTP): This field indicates the largest average power consumed by the NVM subsystem over a 10 second period in this power state with the workload indicated in the Active Power Workload field. The power in Watts is equal to the value in this field multiplied by the scale indicated in the Active Power Scale field. A value of 0h indicates Active Power is not reported.</td></tr>
<tr><td>159:152</td><td>Reserved</td></tr>
<tr><td>151:150</td><td>Idle Power Scale (IPS): This field indicates the scale for the Idle Power field.
<table>
<tr><th>Value</th><th>Definition</th></tr>
<tr><td>00b</td><td>Not reported for this power state</td></tr>
<tr><td>01b</td><td>0.0001 W</td></tr>
<tr><td>10b</td><td>0.01 W</td></tr>
<tr><td>11b</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>149:144</td><td>Reserved</td></tr>
</table>

32 Bytes Power State

Figure 248: Identify – Power State Descriptor Data Structure

Bits	Description
143:128	**Idle Power (IDLP):** This field indicates the typical power consumed by the NVM subsystem over 30 seconds in this power state when idle (i.e., there are no pending commands, register accesses, background processes, sanitize operation, nor device self-test operations). The measurement starts after the NVM subsystem has been idle for 10 seconds. The power in Watts is equal to the value in this field multiplied by the scale indicated in the Idle Power Scale field. A value of 0h indicates Idle Power is not reported. Refer to section 8.4. Note: This value may be used by hosts to manage power versus resume latency. Platform and form factor specifications may have additional power measurement and reporting requirements that are outside the scope of this specification.
127:125	Reserved
124:120	**Relative Write Latency (RWL):** This field indicates the relative write latency associated with this power state. The value in this field shall be less than the number of supported power states (e.g., if the controller supports 16 power states, then valid values are 0 through 15). A lower value means lower write latency.
119:117	Reserved
116:112	**Relative Write Throughput (RWT):** This field indicates the relative write throughput associated with this power state. The value in this field shall be less than the number of supported power states (e.g., if the controller supports 16 power states, then valid values are 0 through 15). A lower value means higher write throughput.
111:109	Reserved
108:104	**Relative Read Latency (RRL):** This field indicates the relative read latency associated with this power state. The value in this field shall be less than the number of supported power states (e.g., if the controller supports 16 power states, then valid values are 0 through 15). A lower value means lower read latency.
103:101	Reserved
100:96	**Relative Read Throughput (RRT):** This field indicates the relative read throughput associated with this power state. The value in this field shall be less than the number of supported power states (e.g., if the controller supports 16 power states, then valid values are 0 through 15). A lower value means higher read throughput.
95:64	**Exit Latency (EXLAT):** This field indicates the maximum exit latency in microseconds associated with exiting this power state. A value of 0h indicates Exit Latency is not reported.
63:32	**Entry Latency (ENLAT):** This field indicates the maximum entry latency in microseconds associated with entering this power state. A value of 0h indicates Entry Latency is not reported.
31:26	Reserved
25	**Non-Operational State (NOPS):** This bit indicates whether the controller processes I/O commands in this power state. If this bit is cleared to '0', then the controller processes I/O commands in this power state. If this bit is set to '1', then the controller does not process I/O commands in this power state. Refer to section 8.4.1.
24	**Max Power Scale (MXPS):** This bit indicates the scale for the Maximum Power field. If this bit is cleared to '0', then the scale of the Maximum Power field is in 0.01 Watts. If this bit is set to '1', then the scale of the Maximum Power field is in 0.0001 Watts.
23:16	Reserved
15:00	**Maximum Power (MP):** This field indicates the sustained maximum power consumed by the NVM subsystem in this power state. The power in Watts is equal to the value in this field multiplied by the scale specified in the Max Power Scale bit. A value of 0h indicates Maximum Power is not reported. Refer to section 8.4. Note: This value is intended to provide an approximate guideline for hosts to manage power versus performance. Platform and form factor specifications may have additional power measurement and reporting requirements that are outside the scope of this specification.

Supported Power Features Via the Identify Command

Within this identify command, users can view the number of power states a controller supports along with the details of each power state. If the controller supports Autonomous Power State transition (apsta), the value will be set to 1.

Figure 247: Identify – Identify Controller Data Structure

Bytes	O/M [1]	Description
265	O	**Autonomous Power State Transition Attributes (APSTA):** This field indicates the attributes of the autonomous power state transition feature. Refer to section 8.4.2. Bits 7:1 are reserved. Bit 0 if set to '1', then the controller supports autonomous power state transitions. If cleared to '0', then the controller does not support autonomous power state transitions.

Figure 456: Example Power State Descriptor Table

Power State	Maximum Power (MP)	Entry Latency (ENTLAT)	Exit Latency (EXLAT)	Relative Read Throughput (RRT)	Relative Read Latency (RRL)	Relative Write Throughput (RWT)	Relative Write Latency (RWL)
0	25 W	5 µs	5 µs	0	0	0	0
1	18 W	5 µs	7 µs	0	0	1	0
2	18 W	5 µs	8 µs	1	0	0	0
3	15 W	20 µs	15 µs	2	0	2	0
4	10 W	20 µs	30 µs	1	1	3	0
5	8 W	50 µs	50 µs	2	2	4	0
6	5 W	20 µs	5,000 µs	4	3	5	1

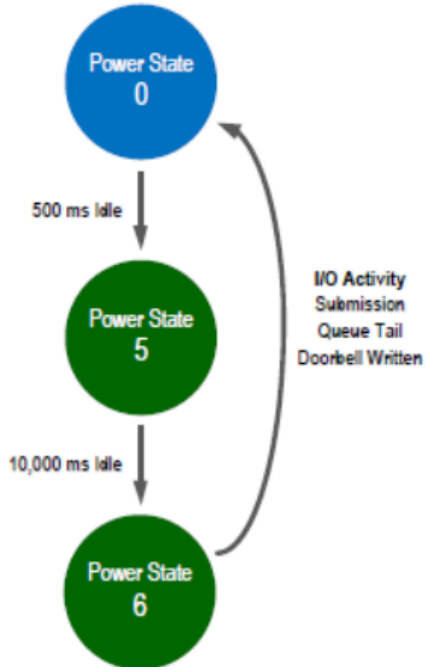
Autonomous Power State transition(APSTA)

Power State Descriptor Table

Power State	Maximum Power	Operational State	Entry Latency	Exit Latency	Relative Read Throughput	Relative Read Latency	Relative Write Throughput	Relative Write Latency
0	25 W	Yes	5 µs	5 µs	0	0	0	0
1	18 W	Yes	5 µs	7 µs	0	0	1	0
2	18 W	Yes	5 µs	8 µs	1	0	0	0
3	15 W	Yes	20 µs	15 µs	2	1	2	1
4	7 W	Yes	20 µs	30 µs	1	2	3	1
5	1 W	No	100 mS	50 mS	-	-	-	-
6	.25 W	No	100 mS	500 mS	-	-	-	-

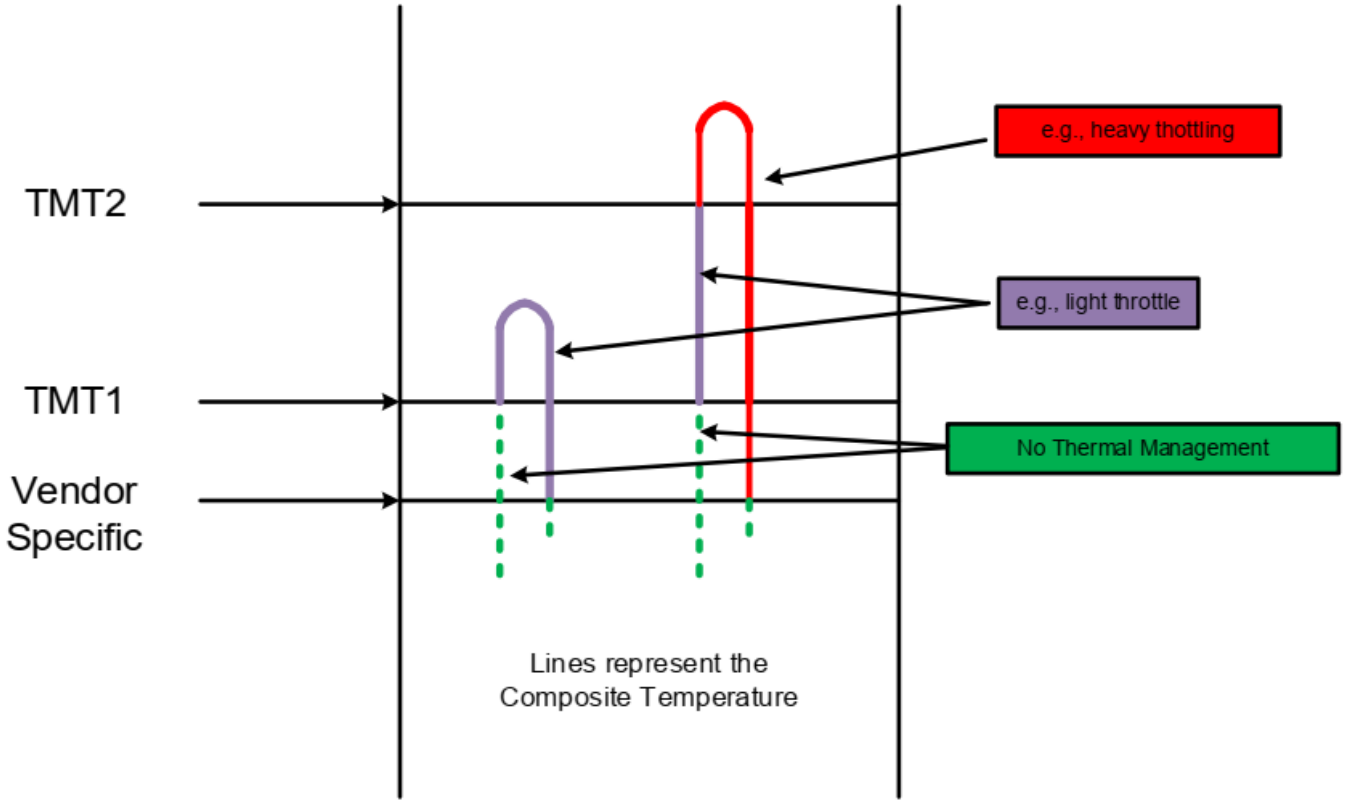
Autonomous Power State Transition Table

Idle Time Prior to Transition	Idle Transition Power State
500 ms	5
500 ms	5
500 ms	5
500 ms	5
500 ms	5
10,000 ms	6
-	-



Thermal Throttle Management

Figure 458: HCTM Example



SSD ,NVME Power State & PCIE Link Power state

SSD Power State	NVMe Power State	PCIe Link
Active/Idle	PS0	L0/L1
Active/Idle (Light Throttle)	PS1	L0/L1
Active/Idle (Heavy Throttle)	PS2	L0/L1
Slumber	PS3	L1/L1.2
Sleep	PS4	L1/L1.2

How PCIe Low Power States and NVMe Technology Features Achieve Near Zero Power Idle Power

NVMe Power State	PCIe Link State	Active / Idle	Power	Exit Latency
PS0	L0 / L0s / L1	Active	100%	No
PS1	L0 / L0s / L1	Active	75%	Very Short
PS2	L0 / L0s / L1	Active	40%	Very Short
PS3	L1 / L1.1 / L1.2	Idle	Low	Moderate
PS4	L1.2	Idle	Extreme Low	Long

Set Features

Figure 271: Set Features – Feature Identifiers

Feature Identifier	Current Setting Persists Across Power Cycle and Reset [2]	Uses Memory Buffer for Attributes	Feature Name
00h	Reserved		
01h	No	No	Arbitration
02h	No	No	Power Management
03h	Yes	Yes	LBA Range Type
04h	No	No	Temperature Threshold
05h	No	No	Error Recovery
06h	No	No	Volatile Write Cache
07h	No	No	Number of Queues
08h	No	No	Interrupt Coalescing
09h	No	No	Interrupt Vector Configuration
0Ah	No	No	Write Atomicity Normal
0Bh	No	No	Asynchronous Event Configuration
0Ch	No	Yes	Autonomous Power State Transition
0Dh	No [3]	No [4]	Host Memory Buffer
0Eh	No	Yes	Timestamp
0Fh	No	No	Keep Alive Timer
10h	Yes	No	Host Controlled Thermal Management
11h	No	No	Non-Operational Power State Config
12h	Yes	No	Read Recovery Level Config
13h	No	Yes	Predictable Latency Mode Config
14h	No	No	Predictable Latency Mode Window
15h	No	No	LBA Status Information Report Interval
16h	No	Yes	Host Behavior Support
17h	Yes	No	Sanitize Config
18h	No	No	Endurance Group Event Configuration
19h to 77h	Reserved		
78h to 7Fh	Refer to the NVMe Management Interface Specification for definition.		
80h to BFh			Command Set Specific (Reserved)

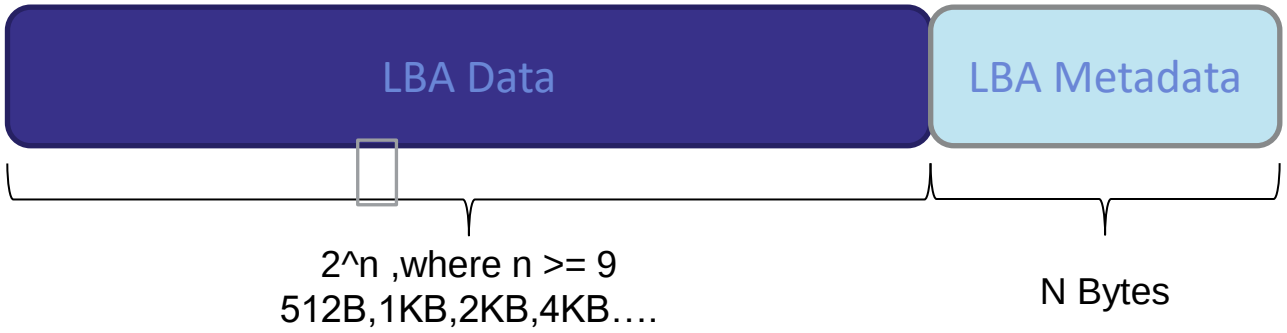
Set Features

Figure 274: Power Management – Command Dword 11

Bits	Description
31:08	Reserved
07:05	**Workload Hint (WH):** This field indicates the type of workload expected. This hint may be used by the NVM subsystem to optimize performance. Refer to section 8.4.3 for more details.
04:00	**Power State (PS):** This field indicates the new power state into which the controller is requested to transition. This power state shall be one supported by the controller as indicated in the Number of Power States Supported (NPSS) field in the Identify Controller data structure. If the power state specified is not supported, the controller shall abort the command and should return an error of Invalid Field in Command.

End-to-end Data Protection

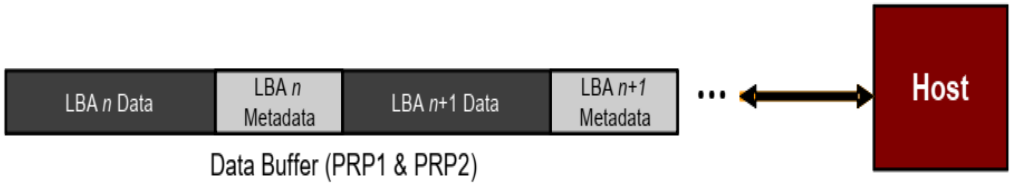
To provide robust data protection from the application to the NVM media and back to the application itself,end-to-end data protection may be used.



Metadata

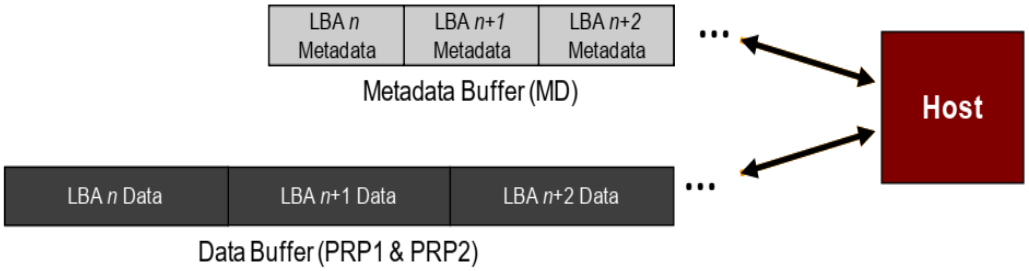
Data Integrity Field (DIF)

Figure 449: Metadata – Contiguous with LBA Data, Forming Extended LBA



Data Integrity Extension (DIX)

Figure 450: Metadata – Transferred as Separate Buffer

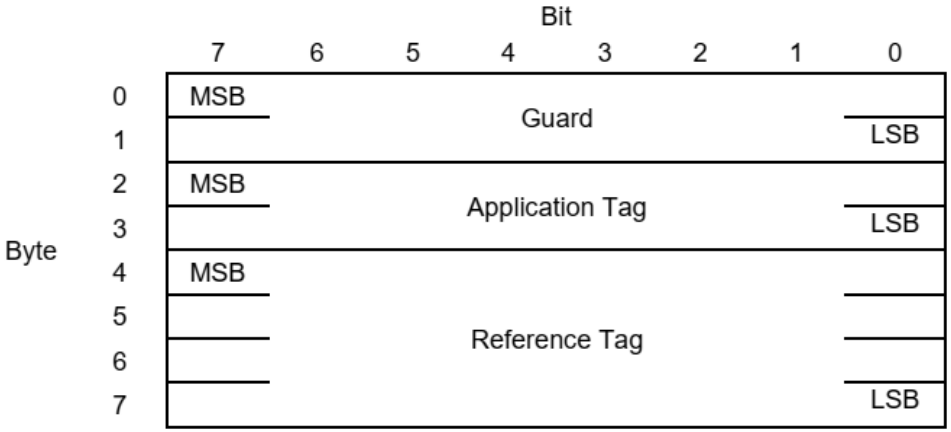


PI

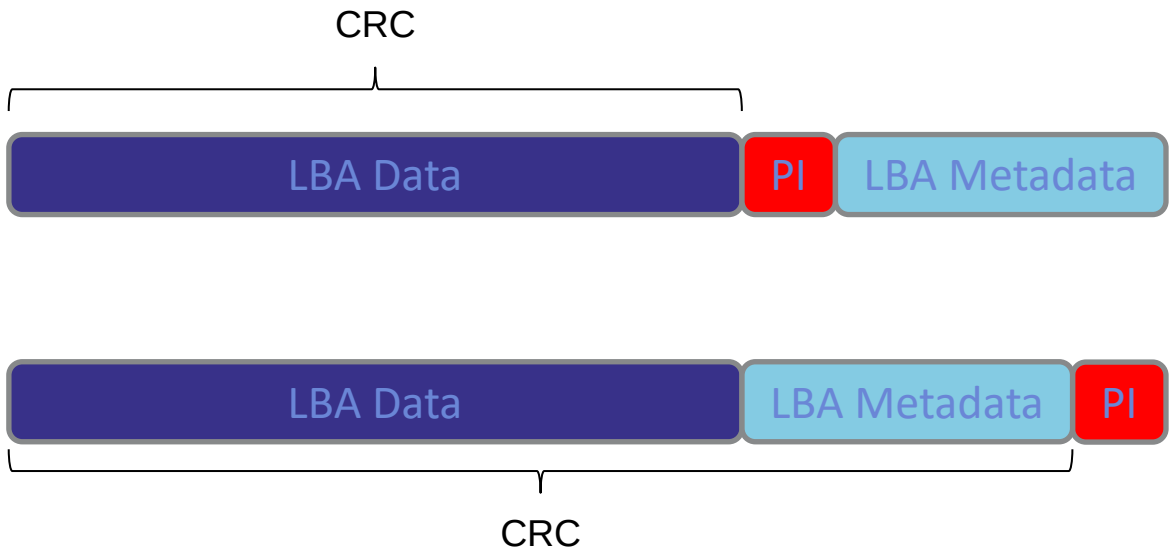
- The Guard field contains a CRC-16 computed over the logical block data.

- The Application Tag is an opaque data field not interpreted by the controller and that may be used to disable checking of protection information .
- The Reference Tag associates logical block data with an address and protects against misdirected or outof-order logical block transfer .

Figure 451: Protection Information Format

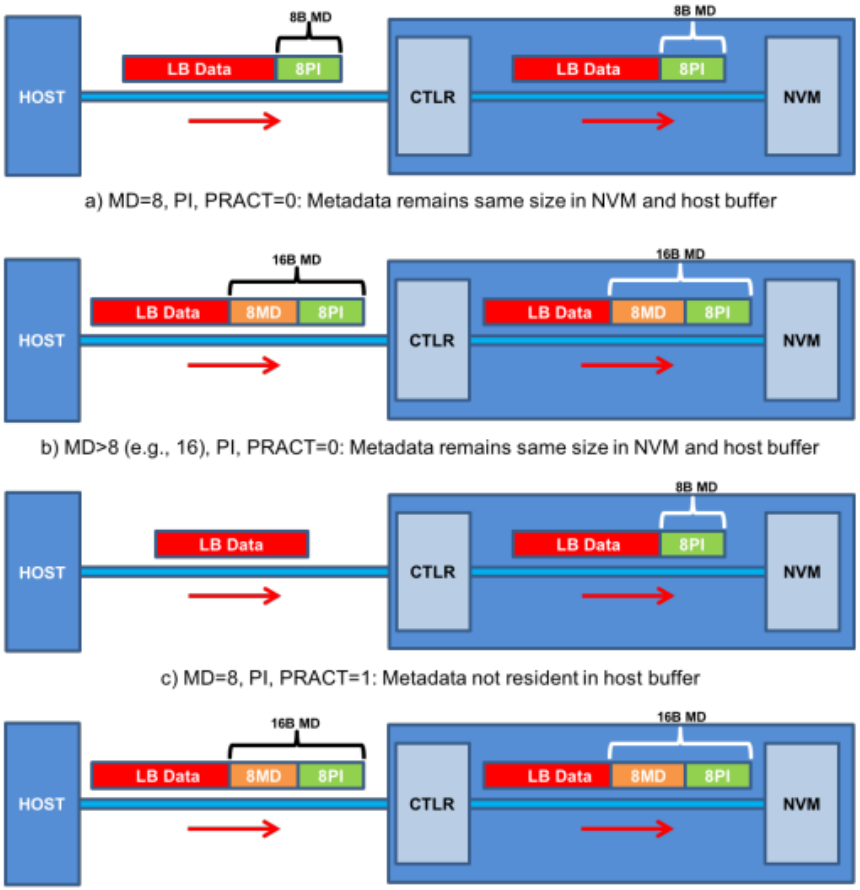


PI



Write Command PI

Figure 452: Write Command Protection Information Processing



a) MD=8, PI, PRACT=0: Metadata remains same size in NVM and host buffer

b) MD>8 (e.g., 16), PI, PRACT=0: Metadata remains same size in NVM and host buffer

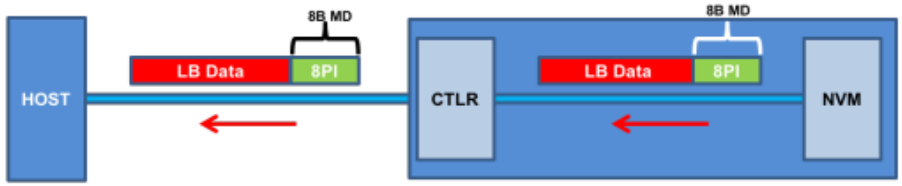
c) MD=8, PI, PRACT=1: Metadata not resident in host buffer

d) MD>8 (e.g., 16), PI, PRACT=1: Metadata remains same size in NVM and host buffer

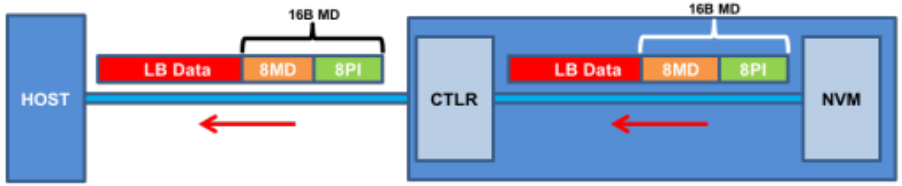
NOTE: In cases (b) and (d) the Protection Information could be before or after the 8 bytes of metadata.

Read Command PI

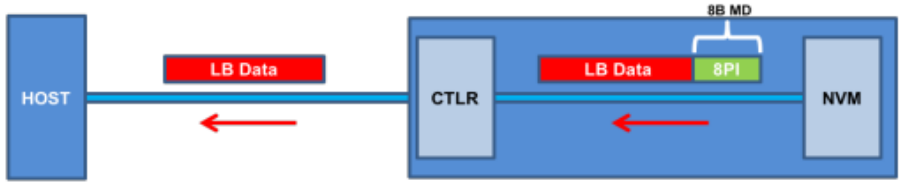
Figure 453: Read Command Protection Information Processing



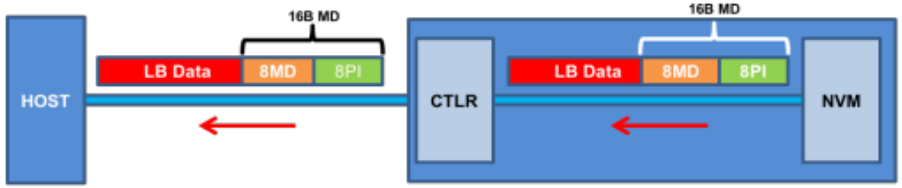
a) MD=8, PI, PRACT=0: Metadata remains same size in NVM and host buffer



b) MD>8 (e.g., 16), PI, PRACT=0: Metadata remains same size in NVM and host buffer



c) MD=8, PI, PRACT=1: Metadata not resident in host buffer



d) MD>8 (e.g., 16), PI, PRACT=1: Metadata remains same size in NVM and host buffer

NOTE: In cases (b) and (d) the PI could be before or after the 8 bytes of metadata.

Enterprise VS Client

Feature	Enterprise Recommended	Client Recommended
I/O Queues	16 to 128	2 to 8
Physically dis-contiguous queues	Design choice	No
Logical block size	4KB	4KB
Interrupt Support	MSI-X	MSI-X
Arbitration	WRR w/ Urgent or Round Robin	Round Robin
AER	Yes	Yes
Firmware Update	Required	Required
End-to-end data protection	Yes	No
SR-IOV support	Yes	No

NVME 1.3

- Device Self Tests
- Sanitize
  1.Block Erase
  2. Crypto Erase
  3. Overwrite
- Namespace Optimal IO Boundary
- Virtualization Management.
- Directives and Streams
- Host Controlled Thermal Management

SSD With no Stream sepatation

SSD With Stream separation

NVME 1.4

- **IO Determinism**

To optimize latency and provide better QoS.

- **Persistent memory Region**

No data is lost when power is off, providing an efficient Non-Volatile storage space for the system. From the current market applications, the most likely application is to store some system log files, as well as some metadata, etc.

- **Multipathing**

Provides a more efficient and convenient data share solution.

IO Determinism

- Provide relatively independent access space.
- Improve IOPS and reduce latency.
- Provide better QoS.

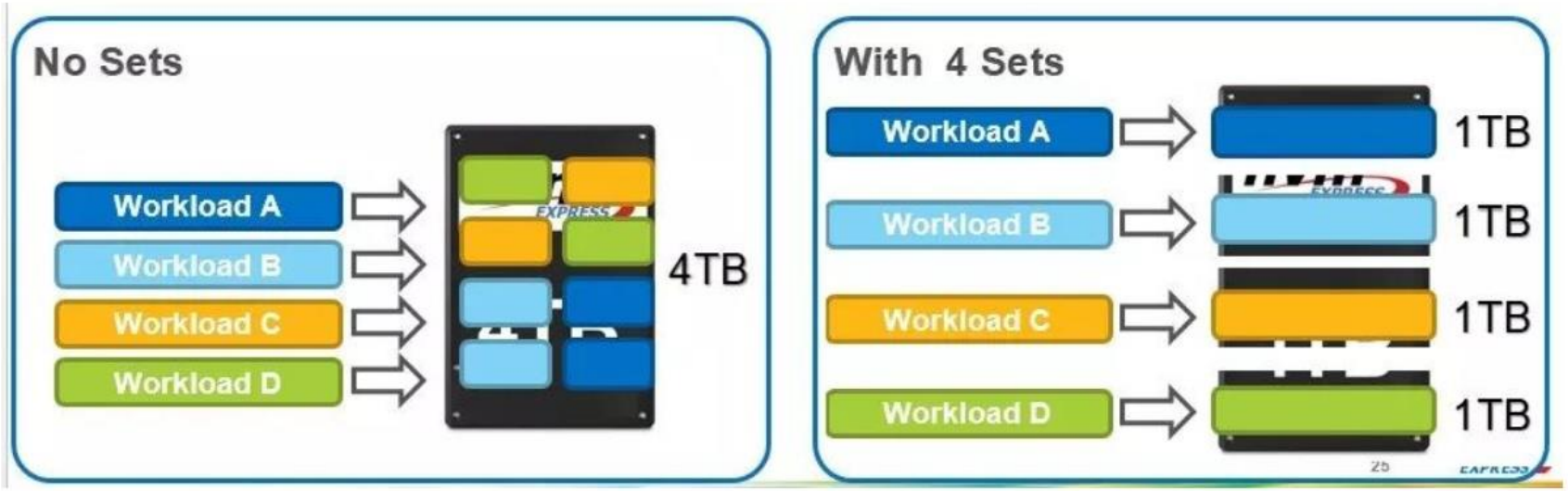
In order to achieve the above points, IO Determinism includes the following parts.

1.NVM Sets.

2.PLM (Predictable Latency Mode).

3.RRL (Read Recovery Level).

NVM SET


No Sets
Workload A
Workload B
Workload C
Workload D
4TB
With 4 Sets
Workload A
Workload B
Workload C
Workload D
1TB
1TB
1TB
1TB

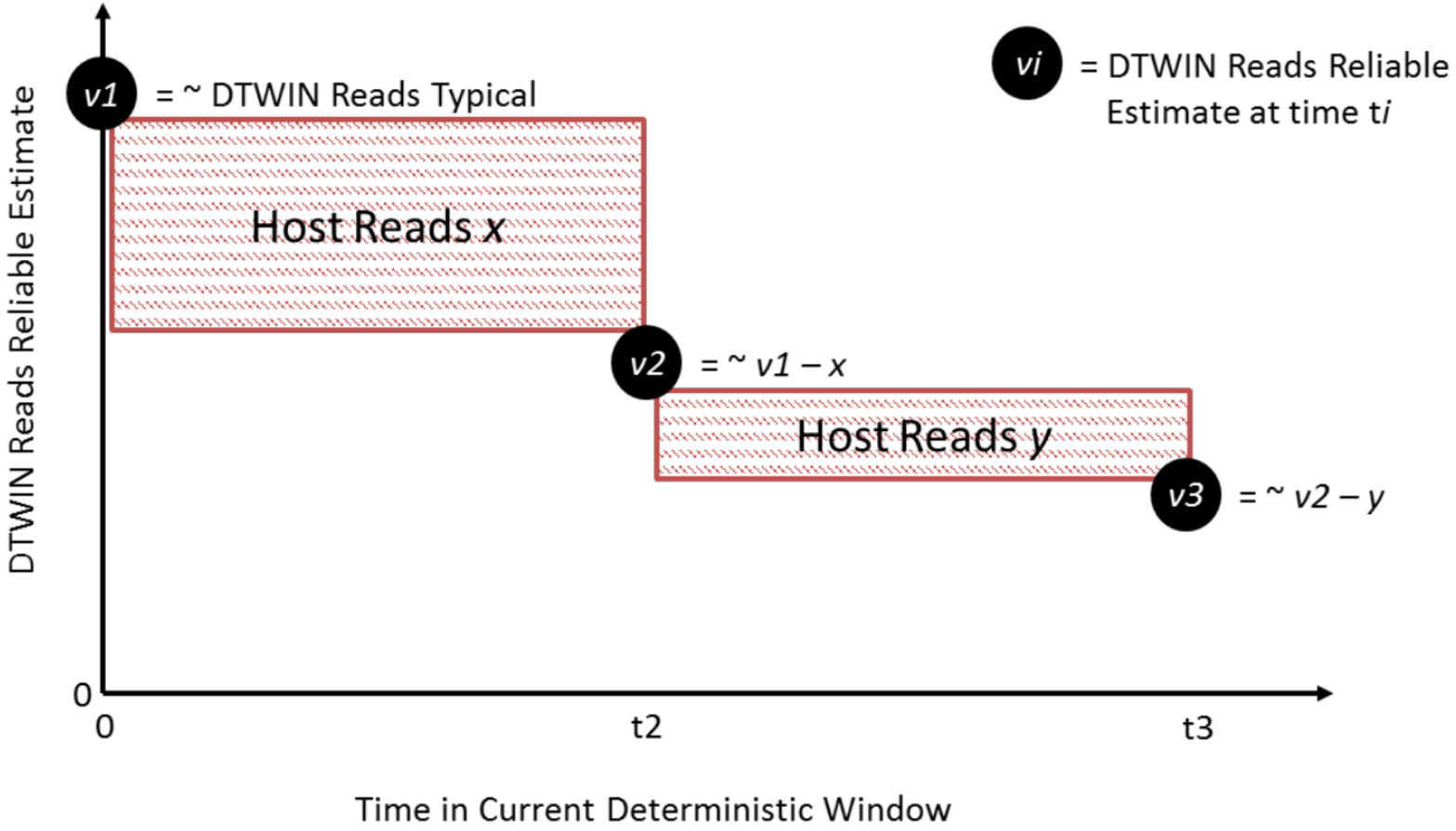
PLM

- The Deterministic Window (DTWIN) is the window of operation during which the NVM Set is able to provide deterministic latency for read and write operations.
- The Non-Deterministic Window (NDWIN) is the window of operation during which the NVM Set is not able to provide deterministic latency for read and write operations as a result of preparing for a subsequent Deterministic Window.

Figure 485: Deterministic and Non-Deterministic Windows

PLM

Figure 487: Typical and Reliable Estimate Example



PLM

- NDWIN Time Minimum Low is the minimum time that the NVM Set remains in the Non-Deterministic Window. The controller may delay completion of a Set Features command requesting a transition to the Deterministic Window until this time is completed. This time does not account for additional host activity in the Non-Deterministic Window.
- NDWIN Time Minimum High is the minimum time that the host should allow the NVM Set to remain in the Non-Deterministic Window after the NVM Set remained in the previous Deterministic Window for DTWIN Time Maximum. This time does not account for additional host activity in the NonDeterministic Window.
- DTWIN Time Maximum is the maximum time that the NVM Set is able to stay in a Deterministic Window.

RRL

Figure 483: Read Recovery Level Overview

Level	O/M	Description
0	O	
1	O	
2	O	
3	O	
4	M	Default
5	O	
6	O	
7	O	
8	O	
9	O	
10	O	
11	O	
12	O	
13	O	
14	O	
15	M	Fast Fail

Maximum Recovery

Decreasing Amount of Recovery

Minimum Recovery

Q&A