SSD Architecture Complexity

Steven Shrader

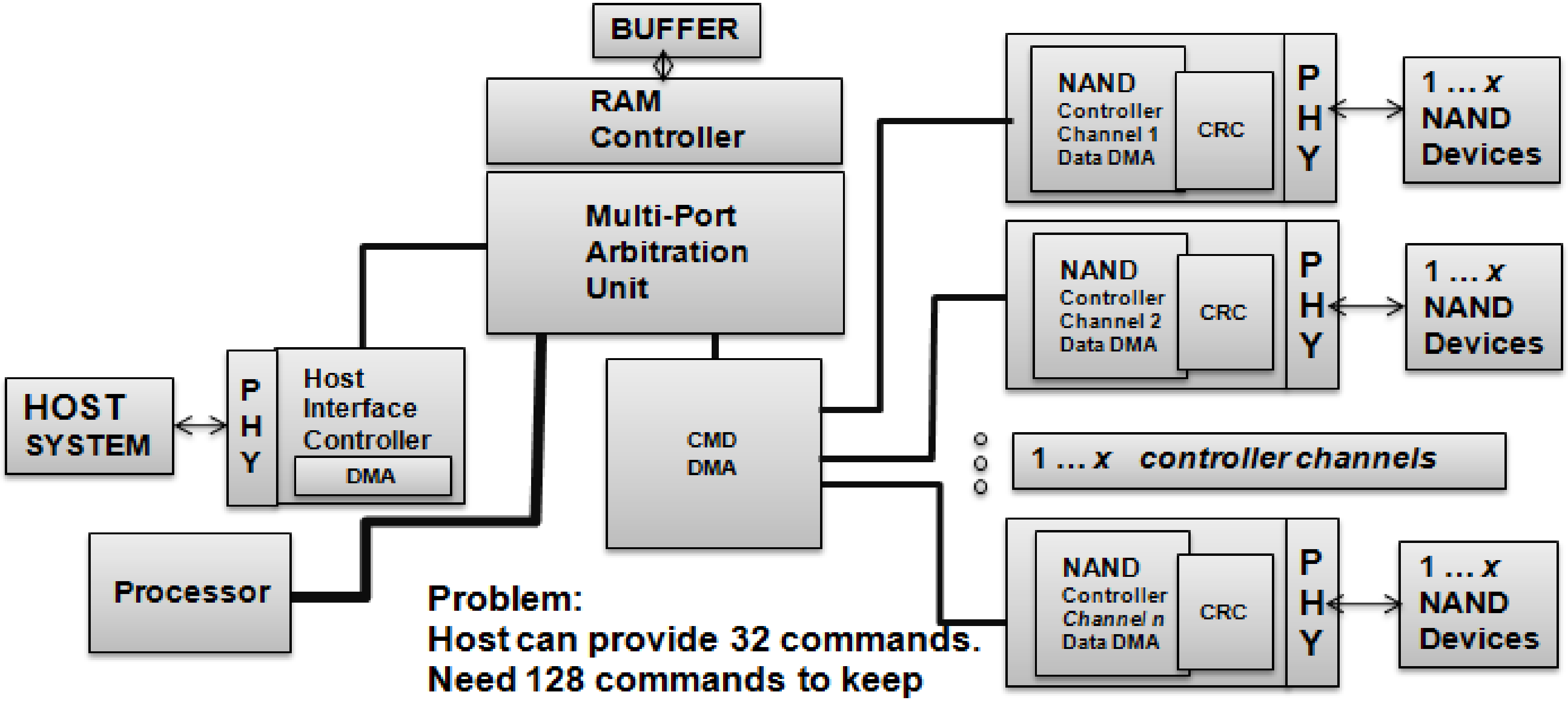
Agenda

- Performance problem with Host interface
- NVM Express performance aspects
- NAND performance aspects
- IP Building Blocks in an Enterprise SSD
- High Performance SSD Architecture components
- High Performance SSD Architecture protocols
- High Performance SSD Architecture Firmware
- The grab bag of performance
- NVM Express performance aspects
- The grab bag of options
- The conclusion?

Performance Problem with Host Interface

- PCIe Gen 3 8 lane performance is about 8GB/s (Max)
- Enterprise random transaction size worst case 4096 bytes
- Best case performance for PCIe Gen 3 8 lane (NVM Express) 1.95M IOPS
- The average length of time for a command 512ns

Performance problem with host interface


BUFFER
RAM Controller
Multi-Port Arbitration Unit
HOST SYSTEM
P H Y
Host Interface Controller
DMA
Processor
CMD DMA
NAND Controller Channel 1 Data DMA
CRC
P H Y
1 … x NAND Devices
NAND Controller Channel 2 Data DMA
CRC
P H Y
1 … x NAND Devices
1 … x controller channels
NAND Controller Channel n Data DMA
CRC
P H Y
1 … x NAND Devices

SATA or SAS
Can provide
32 commands

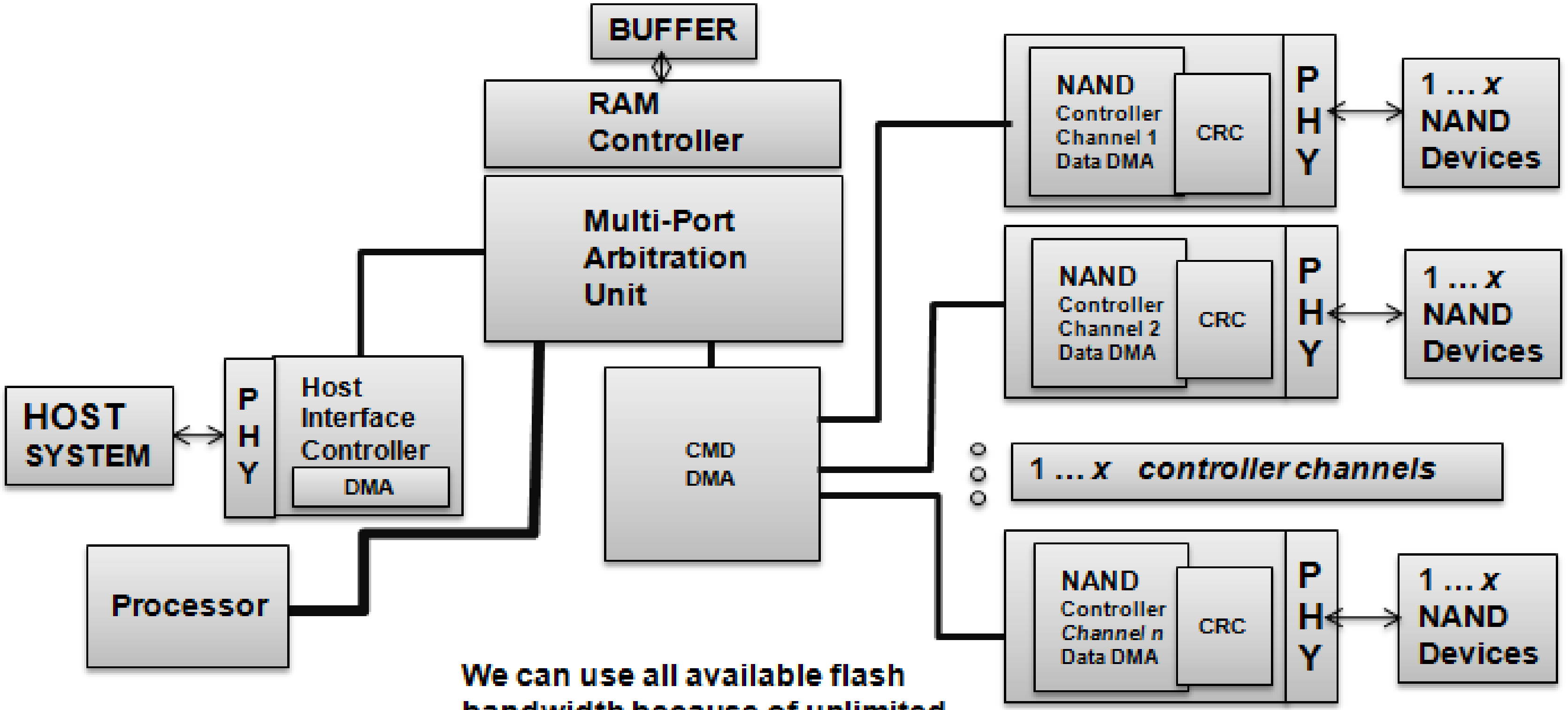
Problem:
Host can provide 32 commands.
Need 128 commands to keep
all flash busy.
At most we can use 25% of our
flash bandwidth
Do we never drive more than 32
flash part because the host
interface does not support it?

16 Flash interfaces
8 chips per flash interface
128 total flash chips
Max performance = 128 commands

NVM Express performance aspects

- As apposed to other disk protocols, there is essentially no limit to the amount of parallelism that can be achieved. This is in line with the need in flash to have a large array all busy at the same time.
- This protocol is truly buffer based transfer protocol. Any buffer of any command can be transferred next (under SSD controller control)
- Multiple command threads implemented in separate queues at different priorities
- NVM Express SSD controller has many degrees of freedom to handle data transfer to gain the best performance in terms of bandwidth and latency

Performance problem with host interface


BUFFER
RAM Controller
Multi-Port Arbitration Unit
HOST SYSTEM
P H Y
Host Interface Controller
DMA
CMD DMA
Processor
NAND Controller Channel 1 Data DMA
CRC
P H Y
1 … x NAND Devices
NAND Controller Channel 2 Data DMA
CRC
P H Y
1 … x NAND Devices
1 … x controller channels
NAND Controller Channel n Data DMA
CRC
P H Y
1 … x NAND Devices

NVM Express Can provide unlimited number of commands

We can use all available flash bandwidth because of unlimited parallelism in the host interface

16 Flash interfaces
16 chips per flash interface
256 total flash chips
Max performance = 256 commands

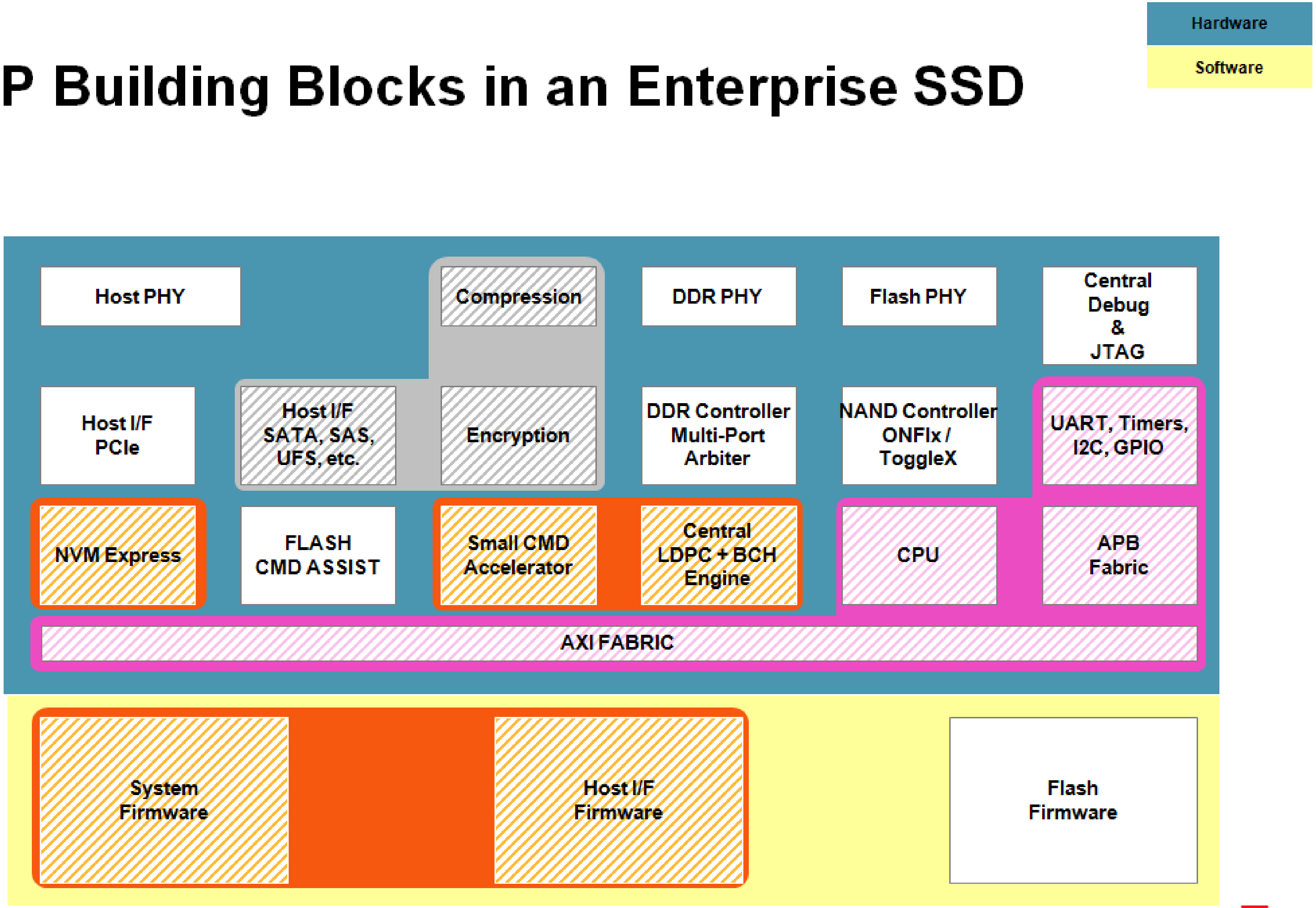
NAND performance aspects 1

- SSD is all about filling the NAND interface with data to 100%
- The faster and smaller the part means that the smaller SSD can have much higher performance with lower memory size that fills the NAND interface
- ONFI has many aspects of performance scheduling
  - Chip selects allow full time multiplexing of transactions
  - LU's on each chip select allow some time multiplexing of transactions
  - Planes on each LU allow interleaving of transactions
  - ONFI allows pipelining of transactions which is a fixed bus interaction between controller and flash part

NAND interface performance

- Best case ONFI3 (200MHz) per interface 400MB/s
- Best case ONFI3 per interface 98K IOPS

IP Building Blocks in an Enterprise SSD



High Performance SSD Architecture components

- Local µP Subsystem
- PCIe
- NVM Express protocol core
- Compression
- Encryption
- Memory (DRAM, RAM)
- Flash channel (parallel data streams)
- ECC – BCH, LDPC, RS, CRC
- Multiple data paths

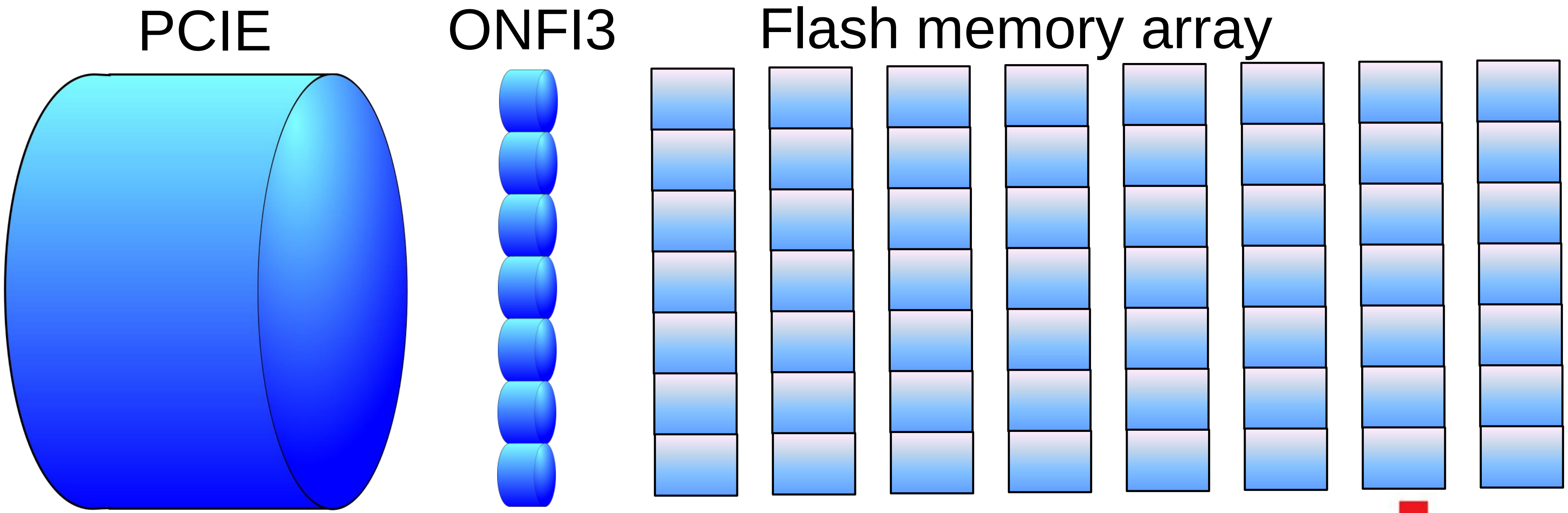
- Buffer staging, locking
- Command ordering
- Data distribution model
- Data encoding model
- Data atomicity method
- Error recovery method
- Data transfer method
- Command tracking/completing
- Etc

High Performance SSD Architecture protocols

- NVM Express
- PCI Express
- NAND Protocols (Like ONFI 3) about 8 of them
- Internal Distribution protocols
- Internal format protocols
- Power loss recovery protocols
- Power control protocols

NAND interface performance

- Number of ONFI3 busses to meet PCIe performance 20 busses
- Number of chips on a ONFI3 interface 8 chips
- Minimum number of commands to keep all chips busy 160 commands



High Performance SSD Architecture Firmware

- Software structure
- Method of communication with hardware (LLD)
- Complement of hardware in terms of functionality
- Must be able to run in parallel with hardware function

The grab bag of performance

- One really important part of Architecture/Design is performance tuning
- While this is important, it is usually a task that is performed at the end
- Sometimes you can make big changes in performance when most everything is directly controlled in software
- Much higher performance interfaces will make controlling all aspects of the controller in software not feasible
- Hardware acceleration for standard read/writes is going to be needed in order to meet interface performance requirements
- Expect no more that about 10% change when tuning performance in higher performance systems of > 1M IOPS

The grab bag of options

- Compatibility in the host interface (various levels of support for options)
- Compatibility in the NAND interface
- Compression
- Encryption
- Level of ECC support
- Write atomicity level
- Level of sudden lower loss support
- Level of flash memory part support

The conclusion?

- Highly complex from a high speed interface perspective (commands in parallel)
- All functions affect and/or interact with other functions making the total number of possible outcomes from SSD development difficult to control
- Complex performance models based upon used options
- Large number of options for interfaces and combination of interfaces (PCIe NVM Express, SATA, SAS…)
- Large number of NAND interface types (Raw, ONFI1, ONFI2, ONFI3, E2NAND, clear NAND, error free NAND…)
- We have ways of making this all work