LightNVM: The Open-Channel SSD Subsystem

Matias Bjørling / Engineering Manager / CNEX LABS
I/O Predictability and Isolation

0% writes and latency is consistent

20% writes makes big impact on read latency

I/O Performance is unpredictable due to writes being buffered

50% writes can make SSDs as slow as spinning drives...
Log-on-log, Indirection, and Narrow I/O

Even if Writes and Reads does not collide from application
Indirection and loss of information due to a Narrow I/O interface

Log-on-Log

Write Indirection & Lost State

FTL-like implementation at multiple layers

Not able to align data on media = Write amplification increase + extra GC

Read/Write Interface makes Data placement + Buffering = Best Effort

Host does not know SSD state due to the narrow I/O Interface
Solid-State Drives and Non-Volatile Media

Solid-State Drive

Host Interface

Responsibilities
- Flash Translation Layer
- Media Error Handling
- Media Retention Management

Media Controller

Parallel Units
- Channel X
- Channel Y

Transform R/W/E to R/W

Manage Media Constraints
- ECC, RAID, Retention

Read/Write

Read/Write/Erase

NAND

Read (50-100us)
Write (1-10ms)
Erase (3-15ms)

Tens of Parallel Units!
New Storage Interface that provides

- Predictable I/O
- I/O Isolation
- Reduces Write Amplication
- Removal of multiple log-structured data structures
- Intelligent data placement and I/O scheduling decisions
- Make the host aware of the SSD state to make those decisions
Outline

1. Physical Page Addressing (PPA) I/O Interface
2. The LightNVM Subsystem
3. pblk: A host-side Flash Translation Layer for Open-Channel SSDs
4. Demonstrate the effectiveness of this interface
Physical Page Addressing (PPA) Interface

- Expose geometry of the SSD
  - Logical/Physical geometry
  - Performance
  - Controller functionalities

- Hierarchical Address Space
  - Encode geometry into the address space

- Vector I/Os
  - Read/Write/Erase

Up to the SSD vendor

Encode parallel units into the address space

Efficient access to the given this new address space
LightNVM Architecture

1. NVMe Device Driver
   - Detection of OCSSD
   - Implements PPA interface

2. LightNVM Subsystem
   - Generic layer
   - Core functionality
   - Target management (e.g., pblk)

3. High-level I/O Interface
   - Block device using pblk
   - Application integration with liblightnvm

Diagram:
- User Space
- Kernel Space
- Hardware
- Open-Channel SSD
- File System
- Application(s)
- NVMe Device Driver
- LightNVM Subsystem
- pblk (1)
- Geometry
- Vectored R/W/E (2)
- Scalar Read/Write (optional) (3)
- PPA Addressing
Host-side Flash Translation Layer - pblk

- Mapping table
  - Sector-granularity

- Write buffering
  - Lockless circular buffer
  - Multiple producers
  - Single consumer (Write Thread)

- Error Handling
  - Media write/erase errors

- Garbage Collection
  - Refresh data
  - Rewrite blocks
Experimental Evaluation

- **CNEX Labs Open-Channel SSD**
  - NVMe
  - PCIe Gen3x8
  - 2TB MLC NAND

- **Geometry**
  - 16 channels
  - 8 PUs per channel (Total: 128 PUs)

- **Parallel Unit Characteristics**
  - Page size: 16K + 64B user OOB
  - Planes: 4, Blocks: 1.067, Block Size: 256 Pages

- **Performance**
  - Write: Single PU 47MB/s
  - Read: Single 108MB/s, 280MB/s (64K)

---

Evaluation

- Sanity check & Base
- Interface Flexibility
  - Limit # Active Parallel Write Units
  - Predictable Latency
Base Performance using Vector I/O

Throughput & Latency

Grows with parallelism

RR slightly lower due to scheduling conflicts

Request I/O Size
Limit # Active Writers

- A priori knowledge of workload. E.g., limit to 400MB/s Write
- Limit number of Active PU Writers, and achieve better read latency

Single Read and Write Perf.

Mixed Read/Write

256K Write QD1
256K Read QD16

Write throughput 400MB/s

Write latency increases, and read latency reduces
Predictable Latency

- 4K reads during 64K concurrent writes
- Consistent low latency at 99.99, 99.999, 99.9999
Multi-Tenant Workloads

NVMe SSD

OCSSD

2 Tenants (1W/1R)

4 Tenants (3W/1R)

8 Tenants (7W/1R)
Conclusion

- Physical Page Addressing specification is available
- Linux kernel subsystem for Open-Channel SSDs
  - Initial release in Linux kernel 4.4.
  - User-space library (liblightnvm) support with Linux kernel 4.11.
  - Pblk upstream with Linux kernel 4.12.

- The right time to dive into Open-Channel SSDs
  - More information available at: http://lightnvm.io

- You may visit Lite-On SSD booth # B6 to have a closer look at Open-Channel SSD.