New NVDIMM Architecture Offers a Fast SSD on the Memory Bus

Doug Finke

Xitore, Inc.

Doug.Finke@Xitore.com

Moving Storage Interface to the DRAM Bus Will Be the Next Major Change

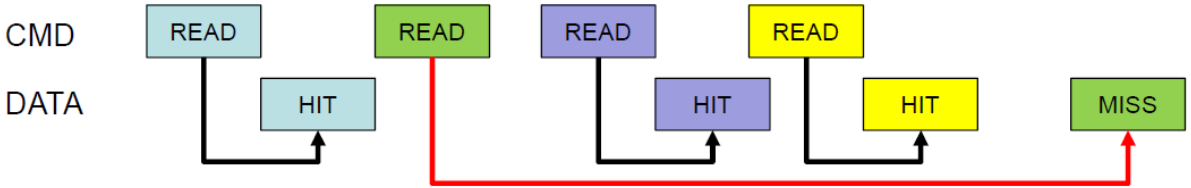
- The bus interface is increasing becoming the largest performance bottleneck in a system.
- Moving persistent storage to the DRAM bus has significant advantages
  - 3X raw speed advantage over NVMe 3.0 x8
  - Opportunity to use byte access instead of block access
  - Use of direct access (DAX) mode for significant reductions in software overheads

Challenges with Existing DRAM Bus

- Current DRAM bus is Deterministic
  - Cannot handle variable access times
  - Cannot handle mixed media
  - Cannot handle execution out-of-order
- Addressing Limitations
  - DDR4 bus limited to 512GB or less

Long Term Solution: NVDIMM-P Protocol for DRAM Bus

- Non-deterministic access with out-of-order completion
- Posted store model using write credits
- Address bus expansion
- Mixed media of various types
- Both cached and non-cached accesses



Example of Out-of-Order Execution with Cached and Non-Cached Accesses

NVDIMM-P Implementation Status

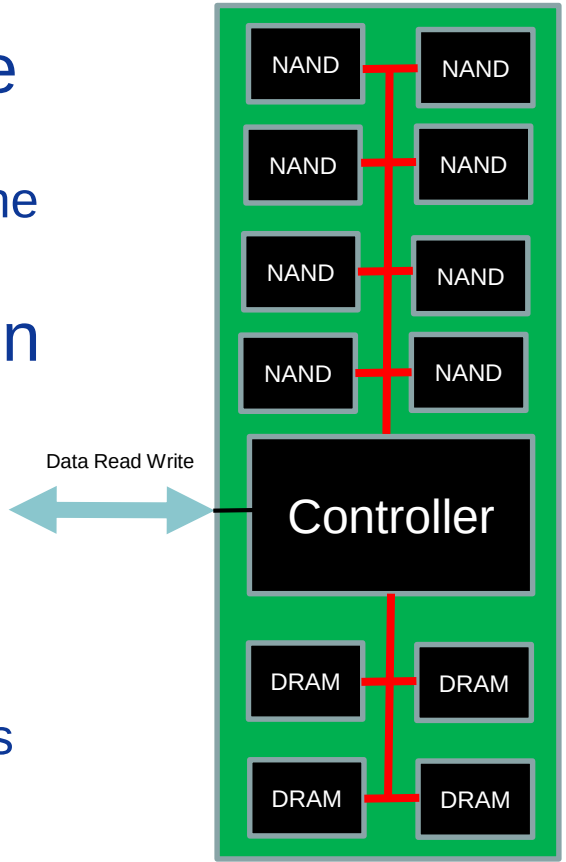
- JEDEC developing full spec to be released in 2018
  - NVDIMM Task Group includes representatives from dozens of companies including all major microprocessor, NAND, and DRAM vendors
  - DDR4: Extension to current spec using two additional control lines (replacing ODT1 & CKE1 on connector)
  - DDR5: Built into main spec
- Will be ideal vehicle for Persistent Memory
- **One catch: Requires new microprocessors to implement new control protocols; Won't work with legacy systems**

Solution: NVDIMM-P with Additional Backwards Compatibility Mode

- Implement flash on the memory bus
- Block mode only in this mode
- Requires a device driver, but no application change
- SSD commands, status, and data passed back and forth as READs and WRITEs on the DRAM bus
- Xitore calls this an NVDIMM-X or NVDIMM-P+

Xitore's NVDIMM-P Implementation with Backwards Compatibility Mode

- Tiered, heterogeneous memory on the same DIMM
  - Both fast and slow memory connected in a cache architecture
- Private on-DIMM memory bus between the fast and slow memories
  - Doesn't tie up the main memory bus for line fetches and cast-outs
- Implements a command queuing capability like HDDs
  - Helps hide latencies when there is a cache miss



How NVDIMM-X Works

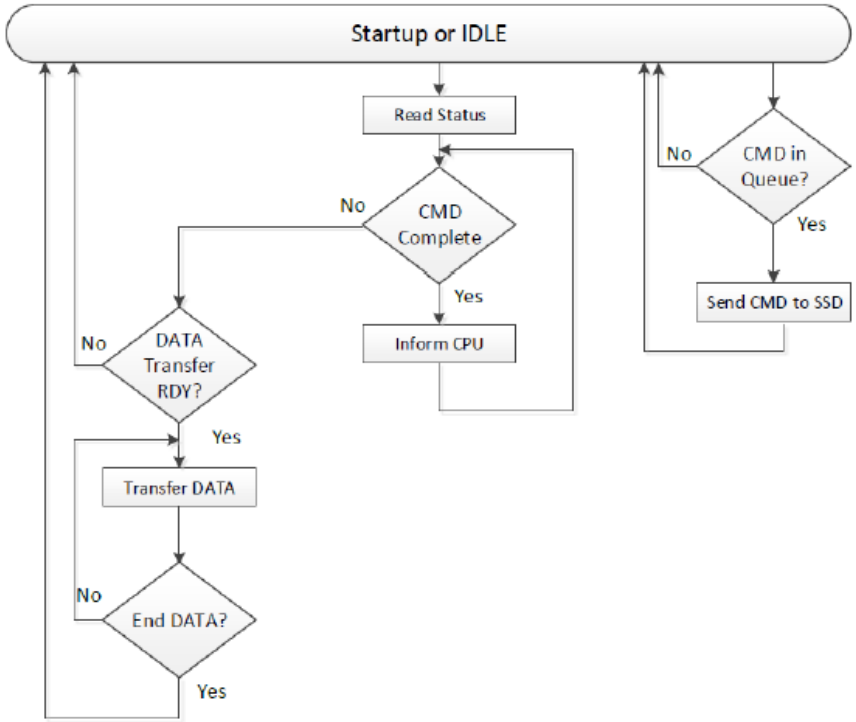
- System configured with new device driver and NVDIMM module
  - No additional App, BIOS, OS or Hardware changes are needed

- Device driver and module communicate commands, status, and data through standard 64 byte DDR4 packets
  - Guarantees deterministic response for legacy microprocessors

- Command queueing is supported with up to 128 commands in the queue.
  - All read and write requests contain a command ID
  - Since address communicated within the 64 byte data packet rather than the standard DDR4 address bus, a full 64 bits of addressability is available.

NVDIMM-X Driver Flowchart

- When device driver issues a command, module will always respond deterministically with a status, i.e. Ready or Not Ready
  - Status of all 128 possible commands communicated at once in 64 byte packet
- When status for a transaction is Ready, data is transferred to the host microprocessor



Summary - Advantages of NVDIMM-X

- Can be used with existing microprocessors
- Solves addressability limit issue
  - Addresses sent over 64 bit data bus using a WRITE command
- Same hardware module can be used as NVDIMM-P in byte mode when appropriate microprocessors become available
- Provides significant performance improvement over NVMe without requiring a change to the microprocessor or application

Thank-You!