SOLID STATE DRIVES

A Tale of $2^1$ Memory(s)

by: Kenneth C. LaMarca
Caleb S. Disbrow
Outline

- History of the SSD
- SSD vs HDD
- Components
- NAND Memory
- Memory Comparison
- SSD Controller
- Alternative Technologies
- Future of the SSD
History

- Origins of the SSD can be seen in similar tech from the 50’s
  - CCROS (Card Caacitor read-only store)
  - Magnetic core memory
- In the 1970’s the first “modern” SSD’s were implemented with semiconductor memory.
- General Instruments produced an electrically alterable ROM (EAROM) which operated similar to modern NAND memory.
- 1988 Digipro revealed a prototype flash SSD based on Intel’s NOR flash memory chips
- 1995 - modern flash memory is used by M-Systems to set the template for flash-based SSDs
- 2006 - SSDs go mainstream with Samsung 2.5-inch 32GB drive using a PATA Interface
SSD vs. HDD

- more expensive per gigabyte
- less power draw, no moving parts, no vibration
- faster read/write speeds
- smaller capacity
- cannot overwrite individual pages
- not affected by magnetism
- 1.8” (standard) form factor vs. a HDD at 2.5” or 3.5”
Price of SSD and HDD trend

- the cost of SSDs have decreased as densities have increased.
- On a cost by gigabyte basis HDDs are still approximately half the cost.
- Due to the intense flooding in Thailand, there were significant HDD shortages causing the price of HDDs to double in that quarter
  - for this reason a comparison of SSD prices and HDD prices are skewed. HDDs are artificially more expensive than due to the floods
Components

- SSD Controller (Processor)
- Interfaces
  - SATA
  - PCI Express
  - USB
  - Fibre Channel (servers)
  - Serial Attached SCSI
- NAND Flash Memory (storage)
- DRAM Cache Memory or Over Provisioning
- (Optional) Battery / Super capacitor
• SLC (Single-level Cell) - data storage in the form of individual flash memory cells
• MLC (Multi-level Cell) - multiple level cells, more bits per transistor
• SLC vs. MLC
  o 3 times faster (sequential read/write)
  o 30% more expensive
• NAND vs. NOR Architecture
  o 4 times faster sequential writes
  o 5 times faster sequential reads
  o 30% less expensive
  o NAND is more dense than NOR
  o NOR is more reliable
NAND Memory

- Represents a Block of memory
- Each row represents a page of memory
- Rows connected via Word line
- Columns connected via Bit line
- Can only write one row at a time, all or nothing, no bit-wise editing
- Each bit is a floating-gate MOSFET
SSD Controller

- has a processor for garbage collection and interfacing with host
- Flash memory Demux and mux
- Buffer manager
- the controller is central to translating the data stored in flash memory
SSD Controller Paging

- the memory controller separates SSDs from other flash memory
- Specifically, how paging is handled
- they cannot overwrite pages, instead they deal with entire blocks when erasing.
- because of this, the SSD will get slower as it reaches capacity
- Hence why garbage collection is important
Trimming

- allows an OS to inform an SSD of no longer used blocks (to be wiped)
- this data is permanently erased
- performance varies based on how the SSD controller handles the TRIM command

1. SSD pages contain no data
2. User writes data to SSD pages
3. User deletes some data. Pages are marked as ‘not In use’ by the host OS, but data remains on SSD.
4. TRIM command tells SSD controller that pages contain invalid data. Pages with invalid data are cleaned.
5. Data is written back to SSD memory cells. The invalid data has been cleaned and data is able to be written to the pages at full speed.
DRAM Cache / Over Provisioning

- an SSD will have either/both
  - DRAM Cache
    - similar to a cache in HDDs
    - cache is used for flipping blocks in garbage collection
  - Over Provisioning
    - flash memory held in provision for garbage collection
    - lose capacity to afford greater performance
    - works in sync with Trim to improve garbage collection
Wear leveling

- Dynamic vs. Static wear leveling
  - shorter life expectancy
  - faster performance
  - less complex design
- Logical Block addresses mapped to physical Flash memory
- Dynamic: Only maps data that is dynamically used, unused data provides no additional wear on the system.
- Static: All data blocks are periodically moved so low usage cells are able to be used by other data.
Error-Correcting Code

- To offset the limitations of MLC NAND memory
- Detect and recover from errors at the bit level (Convolutional codes)
  - Write errors
  - Retention errors - due to loss of charge
  - Read-Disturb errors
- Detect and recover from errors at the block level (Block codes)
  - Examples:
    - Repetition code - looks for repetition in bits and attempts to recreate lost bits in the corrupted block.
    - Hamming code - looks at a parity bit to ensure that the correct number of ones were sent over in the block (even or odd)
**Alternative SSD Technology**

**SATADIMM SSD**
- Draws power from unused DRAM sockets
- Extremely dense
- Used in lower power / high performance server and storage applications

**DDRdrive X1**
- Combines speed, longevity and speed of DRAM with the non-volatility of NAND
- Low capacity, but very high IOPS
The future of SSD technology is heavily reliant on advances in Flash memory. Specifically, transistor size, memory density, and the logistics of handling large scale memory management with Flash Memory has for the most part been worked out. Samsung's VNAND (illustrated here) is a glimpse at the future of Flash memory.
Conclusion

• Based on our findings
  o The future of consumer Storage will heavily rely on SSDs
  o SSDs will not replace HDDs as a consumer mass storage solution in the near future
  o An SSD is best used to store programs, boot data, not media.
  o The SSD will probably become a core component in PCs
Citations


