Storage Class Memory, Technology and Use

Rich Freitas
Agenda

- Introduction
- Storage Class Memory Technologies
- Using Storage Class Memories in Systems
- Impact on Systems

Much of the information in this presentation can be found in the IBM Journal special issue on Storage

Definition of **Storage Class Memory** (SCM)

- A new class of data storage/memory devices
  - many technologies compete to be the ‘best’ SCM

**SCM features:**
- Non-volatile (~ 10 years)
- Fast Access times (~ DRAM like)
- Low cost per bit more (DISK like – by 2015)
- Solid state, no moving parts

**SCM blurs the distinction between**
- MEMORY (*fast, expensive, volatile*) and
- STORAGE (*slow, cheap, non-volatile*)
Some Terminology Clarification

- **SCM = Storage Class Memory**
  - SCM describes a *technology*, not a *use*
  - FLASH is an early example of SCM

- **NVRAM = Non Volatile RAM**
  - SCM is one example of NVRAM
  - Other NVRAM types: DRAM+battery or DRAM+disk combos

- **SSD = Solid State Disk**
  - Use of NVRAM for *block oriented* storage applications
System Targets for SCM

Billions!

- Mobile ✓
- Desktop X
- Datacenter ✓
Storage Class Memory

A solid-state memory that **blurs the boundaries** between storage and memory by being **low-cost, fast, and non-volatile**.

- **SCM system requirements for Memory (Storage) apps**
  - No more than 3-5x the **Cost** of enterprise HDD (< $1 per GB in 2012)
  - **<200nsec (<1μsec)** Read/Write/Erase time
  - **>100,000 Read I/O operations** per second
  - **>1GB/sec (>100MB/sec)**
  - **Lifetime** of $10^8$ – $10^{12}$ write/erase cycles
  - 10x lower **power** than enterprise HDD
## Emerging Memory Technologies

Memory technology remains an active focus area for the industry.

### FLASH Extension
- **64Mb FRAM (Prototype)**
  - 0.13um 3.3V

### FRAM
- Ramtron
- Fujitsu
- STMicro
- TI
- Toshiba
- Infineon
- Freescale

### MRAM
- IBM
- Infineon
- Philips
- STMicro
- Toshiba
- HP
- Hitachi
- NEC
- Honeywell
- Phillips

### PCRAM
- Ovonyx
- IBM
- Samsung
- Elpida
- Infineon
- Toshiba
- NEC
- Sony
- Fujitsu
- Renesas
- Samsung
- Oki
- Hynix
- Celis

### RRAM
- IBM
- Sharp
- Unity
- Samsung
- NEMI
- Hitachi
- Philips

### Solid Electrolyte
- Axon
- Infineon
- BAE
- Spansion
- Samsung
- TFE
- MEC
- Zettacore
- Roltronics
- Nanolayer

### Polymer/Organic
- Axon
- Infineon
- Sharp
- Samsung
- TFE
- MEC
- Zettacore
- Roltronics
- Nanolayer
Papers presented at
- Symposium on VLSI Technology
- IEDM (Int. Electron Devices Meeting)

Research interest
Industry interest in non-volatile memory
Candidate device technologies

- Improved Flash

- FeRAM (Ferroelectric RAM)
  - FeFET

- MRAM (Magnetic RAM)
  - Racetrack memory

- RRAM (Resistive RAM)
  - Organic & polymer memory

- Solid Electrolyte

- PC-RAM (Phase-change RAM)
What is Flash?

- Based on MOS transistor
- Transistor gate is redesigned
  - Charge is placed or removed near the “gate”
  - The threshold voltage $V_{th}$ of the transistor is shifted by the presence of this charge
  - The threshold Voltage shift detection enables non-volatile memory function.
FeRAM (Ferroelectric RAM)

ferroelectric material such as lead zirconate titanate (Pb(ZrxTi1-x)O) or PZT

metallic electrodes

need select transistor – “half-select” perturbs

- perovskites (ABO₃) = 1 family of FE materials
- destructive read → forces need for high write endurance
- inherently fast, low-power, low-voltage
- first demonstrations ~1988

[Sheikholeslami:2000]
MRAM (Magnetic RAM)

- Reading a bit
- Writing "1"
- Writing "0"

MTJ MagRAM promises:
- density of DRAM
- speed of SRAM
- non-volatility

Graph:
- TMR (%)
- Field (Oe)
- MgO barrier
- CoFe/CoFeB free layer
- 350% TMR ($V = 0$)
(2005)

Gallagher:2006
Magnetic Racetrack Memory

- Data stored as pattern of magnetic domains in long nanowire or “racetrack” of magnetic material.
- Current pulses move domains along racetrack.
- Use deep trench to get many (10-100) bits per $4F^2$.
RRAM (Resistive RAM)

- Numerous examples of materials showing hysteretic behavior in their I-V curves
- Mechanisms not completely understood, but major materials classes include
  - metal nanoparticles (?) in organics
    - could they survive high processing temperatures?
  - oxygen vacancies (?) in transition-metal oxides
    - forming step sometimes required
    - scalability unknown
    - no ideal combination yet found of
      - low switching current
      - high reliability & endurance
      - high ON/OFF resistance ratio
  - metallic filaments in solid electrolytes
Solid Electrolyte

Resistance contrast by forming a metallic filament through insulator sandwiched between an inert cathode & an oxidizable anode.

- Ag and/or Cu-doped $\text{Ge}_x\text{Se}_{1-x}$, $\text{Ge}_x\text{S}_{1-x}$ or $\text{Ge}_x\text{Te}_{1-x}$
- Cu-doped $\text{MoO}_x$
- Cu-doped $\text{WO}_x$
- RbAg$_4$I$_5$ system

Advantages

- Program and erase at very low voltages & currents
- High speed
- Large ON/OFF contrast
- Good endurance demonstrated
- Integrated cells demonstrated

Issues

- Retention
- Over-writing of the filament
- Sensitivity to processing temperatures (for GeSe, < 200°C)
- Fab-unfriendly materials (Ag)
Candidate device technologies

- **Improved Flash**
  - little improvement expected in write endurance or speed

- **FeRAM** – commercial product but difficult to scale!
  - **FeFET** – old concept, with many roadblocks

- **MRAM** – commercial product, also difficult to scale!
  - **Racetrack memory** – new concept w/ promise, still at point of early basic physics research

- **RRAM** – few demos showing real CMOS integration
  - **Organic & polymer memory** – temperature compatibility?

- **Solid Electrolyte** – shows real promise if tradeoff between retention & overprogramming can be solved...

- **PC-RAM** (Phase-change RAM)
History of Phase-change memory

- late 1960’s – Ovshinsky shows reversible electrical switching in disordered semiconductors
- early 1970’s – much research on mechanisms, but everything was too slow!
Phase-change RAM

PCRAM
“programmable resistor”

Access device
(transistor, diode)

Potential headache:
High power/current
→ affects scaling!

Potential headache:
If crystallization is slow
→ affects performance!

“SET” pulse

“RESET” pulse

Voltage
temperature

Word-line

Bit-line

T_{melt}

T_{cryst}

time
Cost competition between IC, magnetic and optical devices comes down to effective areal density.

**Density is key**

**Effective Areal Density**

<table>
<thead>
<tr>
<th>Device</th>
<th>Critical feature-size $F$</th>
<th>Area ($F^2$)</th>
<th>Density (Gbit /sq. in)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hard Disk</td>
<td>100 nm (MR width)</td>
<td>0.5</td>
<td>125</td>
</tr>
<tr>
<td>DRAM</td>
<td>90 nm (half pitch)</td>
<td>8.0</td>
<td>10</td>
</tr>
<tr>
<td>NAND (2 bit)</td>
<td>90 nm (half pitch)</td>
<td>3.0</td>
<td>26</td>
</tr>
<tr>
<td>NAND (1 bit)</td>
<td>73 nm (half pitch)</td>
<td>4.7</td>
<td>26</td>
</tr>
<tr>
<td>Blue Ray</td>
<td>210 nm ($\lambda$/2)</td>
<td>1.5</td>
<td>12</td>
</tr>
</tbody>
</table>

[Fontana:2004]
3-D stacking

- Stack multiple layers of memory above the silicon in the CMOS back-end

- NOT the same as 3-D packaging of multiple wafers requiring electrical vias through-silicon

- Issues with temperature budgets, yield, and fab-cycle-time

- Still need access device within the back-end
  - re-grow single-crystal silicon (hard!)
  - use a polysilicon diode (but need good isolation & high current densities)
  - get diode functionality somehow else (nanowires?)
### Paths to ultra-high density memory

At the 32nm node in 2013, MLC NAND Flash (already $M=2 \rightarrow 2F^2$ !) is projected* to be at...

<table>
<thead>
<tr>
<th>Factor</th>
<th>Density (Gb/cm²)</th>
<th>Product (GB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2x</td>
<td>43</td>
<td>32</td>
</tr>
<tr>
<td>4x</td>
<td>86</td>
<td>64</td>
</tr>
<tr>
<td>16x</td>
<td>344</td>
<td>256</td>
</tr>
<tr>
<td>64x</td>
<td>1376</td>
<td>~1 TB</td>
</tr>
</tbody>
</table>

* 2006 ITRS Roadmap

If we could shrink $4F^2$ by...

- **2x**
  - Density: 43 Gb/cm²
  - Product: 32 GB

- **4x**
  - Density: 86 Gb/cm²
  - Product: 64 GB
  - Example: 4 layers of 3-D ($L=4$)

- **16x**
  - Density: 344 Gb/cm²
  - Product: 256 GB
  - Example: 8 layers of 3-D, 2 bits/cell ($L=8, M=2$)

- **64x**
  - Density: 1376 Gb/cm²
  - Product: ~1 TB
  - Example: 4 layers of 3-D, 4x4 sublithographic ($L=4, N=4^2$)
If you could have SCM, why would you need anything else?

Chart courtesy of Dr. Chung Lam
IBM Research
To be published
IBM Journal R&D
How does SCM compare to existing technologies?

<table>
<thead>
<tr>
<th>Technology</th>
<th>Cost</th>
<th>Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOR FLASH</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAND FLASH</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DRAM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SRAM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HDD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STORAGE CLASS MEMORY</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Challenges with SCM

- **Asymmetric performance**
  - Flash: writes much slower than reads
  - Not as pronounced in other technologies

- **Write endurance**
  - Many SCM technologies “wear out” on writes
  - Flash is an example

- **Bad blocks**
  - Devices are shipped with bad blocks
  - Blocks wear out, etc.
SCM: Flash Storage Design

Host Intf.

CPU

MEMORY

DRAM

Flash Control

Flash

Flash

Flash

Flash

SSD device
Write endurance

- In many SCM technologies writes are cumulatively destructive
- For Flash it is the program/erase cycle
- Current commercial flash and SCM varieties
  - Single level cell (SLC) $\rightarrow 10^5$ writes/cell
  - Multi level cell (MLC) $\rightarrow 10^4$ writes/cell
  - PCM $\rightarrow \sim 10^8$ writes/cell
- Wear leveling
Life-time of SCM devices

In a device that wear out on writes,

\[ T_{\text{life}} = \text{Endurance} \cdot \text{Fill-Time} = E \cdot T_{\text{fill}} \]

\[ T_{\text{fill}} = \frac{C}{B} \quad \text{(Fill-Time)} \]

= time to write all C elements, given write-rate of B

\[ T_{\text{fill}} \sim 1 \text{ sec for DRAM} , \sim 10,000 \text{ seconds for disks} \]

Consider an SLC flash chip with \( C = 8 \text{ M blocks} \), \( E = 10^5 \) and \( B = 600 \text{ b/s} \) (blocks per second where a block = 2 KB)

Without any wear-leveling and looping on one block, \( C = 1 \) (not 8 M blocks) and

\[ T_{\text{life}} = \frac{E}{B} = \frac{10^5}{600} \text{ b/s} = 170 \text{ seconds} \]

(Perfect) Wear-leveling improves \( T_{\text{life}} \) (for 1 block) by the capacity \( C \)

\[ T_{\text{life}} = E \cdot T_{\text{fill}} = E \cdot \frac{C}{B} = 10^5 \cdot 8 \text{ Mblocks}/600 \text{ b/s} = 1.36 \cdot 10^9 \text{ seconds} \]

From \( \sim 3 \text{ minutes} \) to more than 42 years!
Dynamic wear leveling

- Frequently written data – logs, updates, etc.
- Maintain a set of free, erased blocks
- Logical to physical block address mapping
- Write new data of free block
- Erase old location and add to free list.
Static wear leveling

- Infrequently written data – OS data, etc
- Maintain count of erasures per block
- Goal is to keep counts “near” each other
- Simple example: move data from hot block to cold block
  - Write LBA 4
  - D1 → 4
  - 1 now FREE
  - D4 → 1
Lifetime model (more details)

- S are system level management ‘tools’ providing an effective endurance of $E^* = S(E) = E$
  - E is the Raw Device endurance and
  - $E^*$ is the effective Write Endurance

- S includes
  - Static and dynamic wear leveling of efficiency $q < 1$
  - Error Correction and bad block management
  - Overprovisioning
  - Compress, de-duplicate & write elimination…
  - $E^* = E \cdot q \cdot f(\text{error correction}) \cdot g(\text{overprovisioning}) \cdot h(\text{compress})…$
  - With S included, $T_{\text{life}}(\text{System}) = T_{\text{fill}} \cdot E^*$
Paths Forward for SCM

- direct disk replacement with SCM packaged as a SSD
- PCIe card that supports a high bandwidth local or direct attachment to a processor.
- PCIe connected drawer that provides a large scale sharable storage system
- design the storage system or the computer system around SCM from the start
SCM module ‘Specs’ in 2020

- SCM modules may be block oriented storage devices

<table>
<thead>
<tr>
<th></th>
<th>1 TB</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Capacity</strong></td>
<td>1 TB</td>
</tr>
<tr>
<td><strong>Read or Write Access Time</strong></td>
<td>&lt;1 us</td>
</tr>
<tr>
<td><strong>Data Rate</strong></td>
<td>&gt;1GB/s</td>
</tr>
<tr>
<td><strong>Sustained transaction rate</strong></td>
<td>200,000 IOPS</td>
</tr>
<tr>
<td>– 1us + 4K / 1GB/s = 5us</td>
<td></td>
</tr>
<tr>
<td><strong>Sustained bandwidth</strong></td>
<td>800MB/s</td>
</tr>
<tr>
<td>– 4KB/5us = &gt;800MB/s</td>
<td></td>
</tr>
</tbody>
</table>
Basic 2020 Storage Package

- **Nonvolatile memory first level package (FLP) (think DIMM)**
- **FLP controller works in concert with other FLP controllers to manage performance, reliability and power**
  - modules checked by controller
  - Redundancy across first level package
  - Detects and attempts to resolve failures
  - Wear leveling

- **16 modules**
  - 1 TB → 16 TB
  - 800 MB/s → 12.8 GB/s
  - 200 kIOPS → 8 MIOPS
2020 SCM Storage System Package

16 modules per FLP
200 FLPs per 2U drawer
21 2U Drawers per rack
Power & space in the server room

The cache/memory/storage hierarchy is rapidly becoming the bottleneck for large systems. We know how to create MIPS & MFLOPS cheaply and in abundance, but feeding them with data has become the performance-limiting and most-expensive part of a system (in both $ and Watts).

- 5 million HDD
- 16,500 sq. ft. !!
- 22 Mega watts

Extrapolation to 2020
(at 70% CGR \(\rightarrow\) need 2 GIOp/sec)


## Results of Extrapolation

<table>
<thead>
<tr>
<th></th>
<th>Compute centric</th>
<th>Data centric</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>disk</td>
<td>SCM</td>
</tr>
<tr>
<td>Devices</td>
<td>1.3 M Disks</td>
<td>406 K modules</td>
</tr>
<tr>
<td>space</td>
<td>4500 sq.ft.</td>
<td>85 sq. ft.</td>
</tr>
<tr>
<td>power</td>
<td>6,000 kW</td>
<td>41 kW</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>disk</td>
<td>SCM</td>
</tr>
<tr>
<td>Devices</td>
<td>5 M Disks</td>
<td>8 K modules</td>
</tr>
<tr>
<td>space</td>
<td>16,500 sq.ft.</td>
<td>12 sq. ft.</td>
</tr>
<tr>
<td>power</td>
<td>22,000 kW</td>
<td>1 kW</td>
</tr>
</tbody>
</table>

Disk ≡ SCM
Memory/Storage Stack Latency Problem

Century

Storage

- Get data from TAPE (40s)
- Access DISK (5ms)
- Access FLASH (20us)

SCM

- Access PCM (100 – 1000 ns)

Memory

- Get data from DRAM or PCM (60ns)
- Get data from L2 cache (10ns)
- CPU operations (1ns)

Human Scale

- Time in ns
  - 10^10
  - 10^9
  - 10^8
  - 10^7
  - 10^6
  - 10^5
  - 10^4
  - 10^3
  - 10^2
  - 10
  - 1

- Century
  - Human Scale
  - second

IBM Almaden Research center

© 2008 IBM Corporation

Storage Class Memory, Technology and Use
Rich Freitas, IBM Research

July 22, 2008
SCM in a large System

1980
- Logic: CPU
- Memory: RAM
- Active Storage: DISK, TAPE

2008
- Logic: CPU
- Memory: RAM, FLASH SSD
- Active Storage: DISK, TAPE

2013
- Logic: CPU
- Memory: RAM
- Active Storage: SCM, DISK, TAPE
Shift in Systems and Applications

Main Memory:
- **DRAM – Disk – Tape**
  - Cost & power constrained
  - Paging not used
  - Only one type of memory: volatile

Storage:
- Active data on disk
- Inactive data on tape
- SANs in heavy use

Applications:
- Compute centric
- Focus on hiding disk latency

- **DRAM – SCM – Disk – Tape**
  - Much larger memory space for same power and cost
  - Paging viable
  - Memory pools: different speeds, some persistent

- Active data on SCM
- Inactive data on disk/tape
- DAS ??

- Data centric comes to fore
- Focus on efficient memory use and exploiting persistence
Summary

- SCM in the form of Flash and PCM are here today and real. Others will follow.
- SCM will have a significant impact on the design of current and future systems and applications.
Questions