Inventing Our Way Around the Memory Wall

Presented by: Jim Handy, Objective Analysis
Thomas Coughlin, Coughlin Associates
Outline

- Coping with Inefficient Data Movement
- Bringing Persistence Closer to the Processor
- Memory & Storage Interfaces Changing, Growing
- Compute-in-Memory, Computational Storage
- New Algorithms Require New Architectures
- Abandoning the von Neumann Architecture
- Emerging Memories to the Rescue
- Making It All Work Together
- Q&A
Outline

- Coping with Inefficient Data Movement
- Bringing Persistence Closer to the Processor
- Memory & Storage Interfaces Changing, Growing
- Compute-in-Memory, Computational Storage
- New Algorithms Require New Architectures
- Abandoning the von Neumann Architecture
- Emerging Memories to the Rescue
- Making It All Work Together
- Q&A
Data Transfer Has Become The Bottleneck
How Work Gets Done

1. Request Data
2. Receive Data
3. Process Data
4. Write Data
Outline

- Coping with Inefficient Data Movement
- Bringing Persistence Closer to the Processor
- Memory & Storage Interfaces Changing, Growing
- Compute-in-Memory, Computational Storage
- New Algorithms Require New Architectures
- Abandoning the von Neumann Architecture
- Emerging Memories to the Rescue
- Making It All Work Together
- Q&A
Move Persistence Up the Memory/Storage Hierarchy

From Report: Emerging Memories Take Off
Outline

- Coping with Inefficient Data Movement
- Bringing Persistence Closer to the Processor
- Memory & Storage Interfaces Changing, Growing
- Compute-in-Memory, Computational Storage
- New Algorithms Require New Architectures
- Abandoning the von Neumann Architecture
- Emerging Memories to the Rescue
- Making It All Work Together
- Q&A
DRAM: Faster Interfaces and More of ‘em

- FPM
- EDO
- SDRAM
- DDR
- DDR2
- DDR3
- DDR4
- DDR5
- HBM
- HBM2
- HBM2e
- OMI
System-Level Interfaces

PCI → PCIe → NVMe → NVMeoF → CXL → CCIX → OpenCAPI → Gen-Z
Outline

- Coping with Inefficient Data Movement
- Bringing Persistence Closer to the Processor
- Memory & Storage Interfaces Changing, Growing
- Compute-in-Memory, Computational Storage
- New Algorithms Require New Architectures
- Abandoning the von Neumann Architecture
- Emerging Memories to the Rescue
- Making It All Work Together
- Q&A
How Work Gets Done

1. Request Data
2. Receive Data
3. Process Data
4. Write Data
The Network Bottleneck

Performance

Compute

Data Transfer

Storage
Improved Approach

1. Initiate Process
2. Process Data in Place
Compute In Memory/Processing in Memory (PIM)

- Automata: Micron, Natural Intelligence
- TOMI: Ven-Ray
- PIM DPU: UPmem
- Gemini APU: GSI
- Aquabolt-XL: Samsung
- SAPEON: SK hynix
- Various Neural Networks

Goal is to reduce data movement
Computational Storage

- NGD
- ScaleFlux
- Eideticom
- NVXL
- Samsung
- InSpur
- Cohesity
- IBM

Goal is to reduce data movement
Computational Storage Drive (CSD)
Computational Storage Processor

Computational Storage Array
Performance Scales with CSS Count

Fuzzy Search

(POC Unindexed Text Data, Edit Distance = 8, E5-2637v3)

Megabytes per Second

CPU Bound!

~700MB/s

# CSSs

1 8 16 24

3X

100X

Flash Memory Summit 2018

ScaleFlux
Outline

- Coping with Inefficient Data Movement
- Bringing Persistence Closer to the Processor
- Memory & Storage Interfaces Changing, Growing
- Compute-in-Memory, Computational Storage
- New Algorithms Require New Architectures
- Abandoning the von Neumann Architecture
- Emerging Memories to the Rescue
- Making It All Work Together
- Q&A
Tuning Algorithms for Computational Storage & PIM

- **Step 1: Standard application programs, but broken apart**
  - This part’s for the server, that part’s for computational storage

- **Step 2: Optimized routines to improve benefits**
  - Lightly-restructured programs to keep both sides busy

- **Step 3: Altogether new algorithms**
  - Wow! Can we really do that?

- It’s all baby steps
Outline

- Coping with Inefficient Data Movement
- Bringing Persistence Closer to the Processor
- Memory & Storage Interfaces Changing, Growing
- Compute-in-Memory, Computational Storage
- New Algorithms Require New Architectures
- Abandoning the von Neumann Architecture
- Emerging Memories to the Rescue
- Making It All Work Together
- Q&A
Harnessing DRAM’s Internal Peculiarities

- Take advantage of DRAM’s internal weaknesses
  - Uses linear aspects of commodity DRAM chips
- Applies different math: Majority/Not
  - Algorithms must be re-worked
  - Architectures need re-configuring
- In research institutes:
  - ComputeDRAM: Princeton
  - SIMDGRAM: ETH Zurich, U of Ill., etc.
  - Ambit: ETH Zurich, CMU, Microsoft, Nvidia
Neural Nets

- Old idea seeing renewed interest
- Instant Matrix Algebra
  - Somewhat slow because it’s linear
- Simple operation
  - Difficult to set up
- A good accelerator to a standard CPU
- Fits emerging memories well
- Lots of research, but no products
Simplifying AI

\[ V_1 \frac{1}{R_1} + V_2 \frac{1}{R_2} + V_3 \frac{1}{R_3} + V_4 \frac{1}{R_4} \]

64Kb Array = 256 sums of 256 Multiplies EACH!
All in a single cycle.

Should SNIA participate in this?
Outline

- Coping with Inefficient Data Movement
- Bringing Persistence Closer to the Processor
- Memory & Storage Interfaces Changing, Growing
- Compute-in-Memory, Computational Storage
- New Algorithms Require New Architectures
- Abandoning the von Neumann Architecture
- Emerging Memories to the Rescue
- Making It All Work Together
- Q&A
Lots of Emerging Memories…

**MRAM**

**PCM**

**ReRAM**

**FRAM**
What Emerging Memories Can Offer

- **Persistence**
  - Instant On
  - Better for power-loss protection
  - Reduce power consumption

- **Small cell size**
  - Large arrays fit onto the processor die

- **Crosspoint configuration**
  - Fits neural networks well
  - Can store linear values
Hey! We Wrote a Report on These!

- **Emerging Memories Take Off**
  - In-depth coverage of everything in this presentation
  - 231 pages, 155 figures, 36 tables
  - Can be purchased on-line for immediate download

- **Two ways to order:**
  - [https://Objective-Analysis.com/reports/#Emerging](https://Objective-Analysis.com/reports/#Emerging)
Outline

- Coping with Inefficient Data Movement
- Bringing Persistence Closer to the Processor
- Memory & Storage Interfaces Changing, Growing
- Compute-in-Memory, Computational Storage
- New Algorithms Require New Architectures
- Abandoning the von Neumann Architecture
- Emerging Memories to the Rescue
- Making It All Work Together
- Q&A
Standards Are Essential

- SNIA achieved a lot with the NVM programming model
  - Now we need to consider persistent processor caches and registers
- The Computational Storage TWG is well embarked for success
  - Standards and taxonomy are progressing well
  - Processing in Memory (PIM) should follow their lead
    - Perhaps not in SNIA
    - PIM interfaces will need to be standardized as was CXL
- Neural nets may be the next frontier
  - It’s storage, but is it storage?
Outline

- Coping with Inefficient Data Movement
- Bringing Persistence Closer to the Processor
- Memory & Storage Interfaces Changing, Growing
- Compute-in-Memory, Computational Storage
- New Algorithms Require New Architectures
- Abandoning the von Neumann Architecture
- Emerging Memories to the Rescue
- Making It All Work Together
- Q&A
Please take a moment to rate this session.

- Your feedback is important to us.
Coughlin Associates

- https://tomcoughlin.com
- Technical and Market Analysis
- Consulting
- Reports and Newsletter
  - Emerging Memories Report
  - Digital Storage in Media and Entertainment
  - Digital Storage Technology Newsletter
## Objective Analysis: Semiconductor Forecast Accuracy

<table>
<thead>
<tr>
<th>Year</th>
<th>Forecast</th>
<th>Actual</th>
</tr>
</thead>
<tbody>
<tr>
<td>2008</td>
<td>Zero growth at best</td>
<td>-3%</td>
</tr>
<tr>
<td>2009</td>
<td>Growth in the mid teens</td>
<td>-9%</td>
</tr>
<tr>
<td>2010</td>
<td>Should approach 30%</td>
<td>32%</td>
</tr>
<tr>
<td>2011</td>
<td>Muted revenue growth: 5%</td>
<td>0%</td>
</tr>
<tr>
<td>2012</td>
<td>Revenues drop as much as -5%</td>
<td>-2.7%</td>
</tr>
<tr>
<td>2013</td>
<td>Revenues increase nearly 10%</td>
<td>4.9%</td>
</tr>
<tr>
<td>2014</td>
<td>Revenues up 20%+</td>
<td>9.9%</td>
</tr>
<tr>
<td>2015</td>
<td>Revenues up ~10%</td>
<td>-0.2%</td>
</tr>
<tr>
<td>2016</td>
<td>Revenues up ~10%</td>
<td>1.1%</td>
</tr>
<tr>
<td>2017</td>
<td>Revenues up ~20%</td>
<td>22%</td>
</tr>
<tr>
<td>2018</td>
<td>Strong start supports 10+% growth</td>
<td>14%</td>
</tr>
<tr>
<td>2019</td>
<td>Semiconductors down -5%</td>
<td>-12.5%</td>
</tr>
<tr>
<td>2020</td>
<td>Zero growth at best</td>
<td>6.8%</td>
</tr>
<tr>
<td>2021</td>
<td>Revenues grow 6% by remaining flat</td>
<td>26.2%</td>
</tr>
<tr>
<td>2022</td>
<td>Total semi still grows 6%</td>
<td>TBD</td>
</tr>
</tbody>
</table>