Memory Is Critical for Computing
(Recall) Memory Is Critical for Computing

- Performance
- Energy
- Reliability
- Security & Safety
- Cost
- Form Factor
- Quality of Service & Predictability
- Sustainability
- ...
(Recall) Ideal Memory

- Zero access time (latency)
- Infinite capacity
- Zero cost
- Infinite bandwidth (to support multiple accesses in parallel)
- Zero energy
(Recall) How Can We Store Data?

- **Flip-Flops (or Latches)**
  - Very fast
  - Very expensive (one bit costs tens of transistors)

- **Static RAM (SRAM)**
  - Relatively fast
  - Expensive (one bit costs 6+ transistors)

- **Dynamic RAM (DRAM)**
  - Slower, reading and charge leakage destroys content (refresh), needs special process for manufacturing (due to capacitor)
  - Cheap (one bit costs only one transistor plus one capacitor)

- **Other storage technology (flash memory, hard disk, tape)**
  - Much slower, special manufacturing needed, non-volatile
  - Very cheap (one transistor stores many bits or no transistors involved)
Array Organization of Memories

- **Goal:** Efficiently store large amounts of data
  - A memory array (stores data)
  - Address selection logic (selects one row of the array)
  - Readout circuitry (reads data out)

- An M-bit value can be read or written at each unique N-bit address
  - All values can be accessed, but only M-bits at a time
  - Access restriction allows more compact organization
Building Larger Memories

- Requires larger memory arrays
- Large $\rightarrow$ slow
- How do we make the memory large without making it too slow?

Idea: Divide the memory into smaller arrays and interconnect the arrays to input/output buses
- Large memories are hierarchical array structures
- DRAM: Channel $\rightarrow$ Rank $\rightarrow$ Bank $\rightarrow$ Subarrays $\rightarrow$ Mats
(Recall) Generalized Memory Structure
(Recall) Generalized Memory Structure

Cutting Edge: 3D-Stacking of Memory & Logic

Other “True 3D” technologies under development
The DRAM Subsystem
A Top-Down View
DRAM Subsystem Organization

- Channel
- DIMM
- Rank
- Chip
- Bank
- Row/Column
The DRAM Subsystem

“Channel”

DIMM (Dual in-line memory module)

Processor

Memory channel

Memory channel
Breaking down a DIMM (module)

DIMM (Dual in-line memory module)

Side view

Front of DIMM

Back of DIMM
Breaking down a DIMM (module)

DIMM (Dual in-line memory module)

Rank 0: collection of 8 chips

Rank 1

Side view
Rank

Rank 0 (Front)  Rank 1 (Back)

Addr/Cmd  CS <0:1>  Memory channel  Data <0:63>
Breaking down a Rank

Rank 0

<0:63>

Chip 0

<0:7>

Chip 1

<8:15>

... 

<56:63>

Chip 7

Data <0:63>
Breaking down a Chip
Breaking down a Bank

Diagram showing a bank with a row buffer and data transfer to and from it.
Access Address:
(Row 0, Column 0) (Row 0, Column 1) (Row 0, Column 85) (Row 1, Column 0)

Row address 0 → Column address 0 → (Row 0, Column 0)
Row address 0 → Column address 1 → (Row 0, Column 1)
Row address 0 → Column address 85 → (Row 0, Column 85)
Row address 1 → Column address 0 → (Row 1, Column 0)

This view of a bank is an abstraction.

Internally, a bank consists of many cells (transistors & capacitors) and other structures that enable access to cells.
A DRAM Bank Internally Has Sub-Banks

**Figure 1.** DRAM bank organization

Another View of a DRAM Bank

Logical Abstraction

Physical View

More on DRAM Basics & Organization


See Section 2 for comprehensive DRAM Background

In-DRAM Bulk Bitwise Execution Engine

Vivek Seshadri
Microsoft Research India
visesha@microsoft.com

Onur Mutlu
ETH Zürich
onur.mutlu@inf.ethz.ch

DRAM Subsystem Organization

- Channel
- DIMM
- Rank
- Chip
- Bank
- Row/Column
Example: Transferring a cache block

Physical memory space

0xFFFF...F

0x00

0x40

0x00

64B cache block

Mapped to

Channel 0

DIMM 0

Rank 0

Mapped to
Example: Transferring a cache block

Physical memory space

0xFFFF...F

0x40

64B cache block

0x00

Chip 0

Chip 1

... (omitted)

Chip 7

Rank 0

Data <0:63>

<0:7>

<8:15>

<56:63>
Example: Transferring a cache block

Physical memory space

0xFFFF...F

... 0x40

0x00

64B cache block

Row 0 Col 0

Chip 0  Chip 1  Chip 7

Data <0:63>

<0:7>  <8:15>  <56:63>

Rank 0

...
Example: Transferring a cache block

Physical memory space

0xFFFF...F

0x40

0x00

64B cache block

8B

Rank 0

Chip 0

Chip 1

Chip 7

Row 0 Col 0

<0:7>

<8:15>

<56:63>

Data <0:63>

8B
Example: Transferring a cache block

Physical memory space

- 0xFFFF...F
- 0x00
- 0x40
- 0x00

64B cache block

Chip 0
Chip 1
Chip 7

Rank 0

Row 0, Col 1

Data <0:63>

<0:7>, <8:15>, <56:63>
Example: Transferring a cache block

Physical memory space

0xFFFF...F

Chip 0

<0:7>

Row 0
Col 1

<8:15>

<56:63>

Data <0:63>

8B cache block

8B

8B
Example: Transferring a cache block

Physical memory space

A 64B cache block takes 8 I/O cycles to transfer.

During the process, 8 columns are read sequentially.
Memory Technology:
DRAM and SRAM
Memory Technology: DRAM

- **Dynamic random access memory**
- Capacitor charge state indicates stored value
  - Whether the capacitor is charged or discharged indicates storage of 1 or 0
  - 1 capacitor
  - 1 access transistor

- Capacitor leaks through the RC path
  - DRAM cell loses charge over time
  - DRAM cell needs to be refreshed
Accessing a DRAM Cell

- **Wordline**
- **Capacitor**
- **Access Transistor**
- **Enable**
- **Sense Amp**
- **Bitline**
Accessing a DRAM Cell

1. Enable wordline
2. Connects cell to bitline
3. Cell loss charge to bitline
4. Deviation in bitline voltage
5. Enable sense amp
6. Cell loses charge to bitline

[Seshadri+ MICRO’17]
Memory Technology: SRAM

- **Static random access memory**
- Two cross coupled inverters store a single bit
  - Feedback path enables the stored value to stay in the “cell” (as long as powered on)
  - 4 transistors for storage
  - 2 transistors for access
Memory Bank Organization and Operation

- Read access sequence:
  1. Decode row address & drive word-lines
  2. Selected bits drive bit-lines
     - Entire row read
  3. Amplify row data
  4. Decode column address & select subset of row
     - Send to output
  5. Precharge bit-lines
     - For next access
SRAM (Static Random Access Memory)

Read Sequence
1. address decode
2. drive row enable
3. selected bit-cells drive bitlines
   (entire row is read together)
4. differential sensing and column select
   (data is ready)
5. precharge all bitlines
   (for next read or write)

Access latency dominated by steps 2 and 3
Cycling time dominated by steps 2, 3 and 5
- step 2 proportional to $2^m$
- step 3 and 5 proportional to $2^n$
### DRAM (Dynamic Random Access Memory)

- Bit stored as charge on node capacitor (non-restorative)
  - bit cell loses charge when read
  - bit cell loses charge over time

#### Read Sequence
1~3 same as SRAM
4. a “flip-flopping” sense amp amplifies and regenerates the bitline, data bit is mux’ ed out
5. precharge all bitlines

#### Destructive reads

#### Charge loss over time

#### Refresh: A DRAM controller must periodically read each row within the allowed refresh time (10s of ms) such that charge is restored
DRAM vs. SRAM

- **DRAM**
  - Slower access (capacitor)
  - Higher density (1T 1C cell)
  - Lower cost
  - Requires refresh (power, performance, circuitry)
  - Manufacturing requires putting capacitor and logic together

- **SRAM**
  - Faster access (no capacitor)
  - Lower density (6T cell)
  - Higher cost
  - No need for refresh
  - Manufacturing compatible with logic process (no capacitor)
An Aside: Phase Change Memory

- Phase change material (chalcogenide glass) exists in two states:
  - Amorphous: Low optical reflexivity and high electrical resistivity
  - Crystalline: High optical reflexivity and low electrical resistivity

PCM is resistive memory: High resistance (0), Low resistance (1)

DRAM vs. PCM

**DRAM**
- Faster access (capacitor)
- Lower density (capacitor less scalable) → higher cost
- Requires refresh (power, performance, circuitry)
- Manufacturing requires putting capacitor and logic together
- Volatile (loses data at loss of power)
- No endurance problems
- Lower access energy

**PCM**
- Slower access (heating and cooling based "phase change" operation)
- Higher density (phase change material more scalable) → lower cost
- No need for refresh
- Manufacturing requires less conventional processes – less mature
- Non-volatile (does **not** lose data at loss of power)
- Endurance problems (a cell cannot be used after N writes to it)
- Higher access energy
PCM-based Main Memory

- How should PCM-based (main) memory be organized?

- Pure PCM main memory [Lee et al., ISCA’09, Top Picks’10]
  - How to redesign the system to tolerate PCM shortcomings

- Hybrid PCM+DRAM [Qureshi+ ISCA’09, Dhiman+ DAC’09]
  - How to partition/migrate data between PCM and DRAM
Reading: PCM as Main Memory: Idea in 2009

- Benjamin C. Lee, Engin Ipek, Onur Mutlu, and Doug Burger, "Architecting Phase Change Memory as a Scalable DRAM Alternative"
- One of the 13 computer architecture papers of 2009 selected as Top Picks by IEEE Micro. Selected as a CACM Research Highlight. 2022 Persistent Impact Prize.

Architecting Phase Change Memory as a Scalable DRAM Alternative

Benjamin C. Lee†  Engin Ipek†  Onur Mutlu†  Doug Burger†

†Computer Architecture Group
Microsoft Research
Redmond, WA
{blee, ipek, dburger}@microsoft.com

‡Computer Architecture Laboratory
Carnegie Mellon University
Pittsburgh, PA
onur@cmu.edu
Intel Optane Persistent Memory (2019)

- Non-volatile main memory
- Based on 3D-XPoint Technology

[Image of Intel Optane Persistent Memory module]

https://www.storagereview.com/intel_optane_dc_persistent_memory_module_pmm
Charge vs. Resistive Memories

- **Charge Memory** (e.g., DRAM, Flash)
  - Write data by capturing charge $Q$
  - Read data by detecting voltage $V$

- **Resistive Memory** (e.g., PCM, STT-MRAM, memristors)
  - Write data by pulsing current $\frac{dQ}{dt}$
  - Read data by detecting resistance $R$
Promising Resistive Memory Technologies

- **PCM**
  - Inject current to change *material phase*
  - Resistance determined by phase

- **STT-MRAM**
  - Inject current to change *magnet polarity*
  - Resistance determined by polarity

- **Memristors/RRAM/ReRAM**
  - Inject current to change *atomic structure*
  - Resistance determined by atom distance
Phase Change Memory: Pros and Cons

Pros over DRAM
- Better technology scaling (capacity and cost)
- Non volatile → Persistent
- Low idle power (no refresh)

Cons
- Higher latencies: ~4-15x DRAM (especially write)
- Higher active energy: ~2-50x DRAM (especially write)
- Lower endurance (a cell dies after ~10^8 writes)
- Reliability issues (resistance drift)

Challenges in enabling PCM as DRAM replacement/helper:
- Mitigate PCM shortcomings
- Find the right way to place PCM in the system
More on Emerging Memory Technologies

Two-Level Memory/Storage Model

- The traditional two-level storage model is a bottleneck with NVM
- **Volatile** data in memory → a load/store interface
- **Persistent** data in storage → a file system interface
- Problem: Operating system (OS) and file system (FS) code to locate, translate, buffer data become performance and energy bottlenecks with fast NVM stores

Two-Level Store

- Load/Store
- fopen, fread, fwrite, ...
- Operating system and file system

Virtual memory

Address translation

Main Memory

Processor and caches

Persistent (e.g., Phase-Change) Memory

https://www.youtube.com/watch?v=pmLszWGmMGQ&list=PL5Q2soXY2Zi9xidylGxUz7xRPS-wisBN&index=29
More on Memory Technologies

Computer Arch. - Lecture 3b: Memory Systems: Challenges and Opportunities (ETH Zürich, Fall 2020)

1,446 views • Sep 26, 2020

https://www.youtube.com/watch?v=pmLszWGmMGQ&list=PL5Q2soXY2Zi9xidylGxBxUz7xRPS-wisBN&index=29
A Bit on Flash Memory & SSDs

- Flash memory was a very “doubtful” emerging technology
  - for at least two decades

Proceedings of the IEEE, Sept. 2017

Error Characterization, Mitigation, and Recovery in Flash-Memory-Based Solid-State Drives

By Yu Cai, Saugata Ghose, Erich F. Haratsch, Yixin Luo, and Onur Mutlu

Abstract: NAND flash memory is ubiquitous in everyday life today because its capacity has continuously increased and its cost has continuously decreased. This position...

Keywords: Data storage systems; error recovery; fault tolerance; flash memory; reliability; solid-state drives

A Flash Memory SSD Controller

**Fig. 1.** (a) SSD system architecture, showing controller (Ctrl) and chips. (b) Detailed view of connections between controller components and chips.


Lecture on Flash Memory & SSDs

Planar vs. 3D NAND Flash Memory

- **Planar NAND Flash Memory**
  - Scaling: Reduce flash cell size, reduce distance b/w cells
  - Reliability: Scaling hurts reliability
- **3D NAND Flash Memory**
  - Scaling: Increase # of layers
  - Reliability: Not well studied!
SSD Course (Spring 2023)

- **Spring 2023 Edition:**

- **Fall 2022 Edition:**

- **Youtube Livestream (Spring 2023):**
  - [https://www.youtube.com/watch?v=4VTwOMmsnJY&list=PL5Q2soXY2Zi_8qOM5Icpp8hB2SHtm4z57&pp=iAQB](https://www.youtube.com/watch?v=4VTwOMmsnJY&list=PL5Q2soXY2Zi_8qOM5Icpp8hB2SHtm4z57&pp=iAQB)

- **Youtube Livestream (Fall 2022):**
  - [https://www.youtube.com/watch?v=hqLrd-Uj0aU&list=PL5Q2soXY2Zi9BJhenUq4JI5bwhAMpAp13&p=p=iAQB](https://www.youtube.com/watch?v=hqLrd-Uj0aU&list=PL5Q2soXY2Zi9BJhenUq4JI5bwhAMpAp13&p=p=iAQB)

- **Project course**
  - Taken by Bachelor’s/Master’s students
  - SSD Basics and Advanced Topics
  - Hands-on research exploration
  - Many research readings

---

https://www.youtube.com/onurmutlulectures
Lectures on Memory Technologies

- **Computer Architecture, Fall 2020, Lecture 15**
  - Emerging Memory Technologies (ETH, Fall 2020)
  - [https://www.youtube.com/watch?v=AlE1rD9G_YU&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=28](https://www.youtube.com/watch?v=AlE1rD9G_YU&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=28)

- **Computer Architecture, Fall 2020, Lecture 16a**
  - Opportunities & Challenges of Emerging Memory Tech (ETH, Fall 2020)
  - [https://www.youtube.com/watch?v=pmLszWGmMGQ&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=29](https://www.youtube.com/watch?v=pmLszWGmMGQ&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=29)

- **Computer Architecture, Fall 2020, Lecture 3b**
  - Memory Systems: Challenges & Opportunities (ETH, Fall 2020)
  - [https://www.youtube.com/watch?v=Q2FbUxD7GHs&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=6](https://www.youtube.com/watch?v=Q2FbUxD7GHs&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=6)

[https://www.youtube.com/onurmutlulectures](https://www.youtube.com/onurmutlulectures)
Tutorial on Processing in Memory

IEEE CICC Educational Session Talk on Memory-Centric Computing (Prof. Onur Mutlu)

Onur Mutlu
Professor
ETH Zurich & Carnegie Mellon University

23 April 2023

https://www.youtube.com/onurmutlulectures
A Modern Primer on Processing in Memory

Onur Mutlu\textsuperscript{a,b}, Saugata Ghose\textsuperscript{b,c}, Juan Gómez-Luna\textsuperscript{a}, Rachata Ausavarungnirun\textsuperscript{d}

\textit{SAFARI Research Group}

\textsuperscript{a}ETH Zürich
\textsuperscript{b}Carnegie Mellon University
\textsuperscript{c}University of Illinois at Urbana-Champaign
\textsuperscript{d}King Mongkut’s University of Technology North Bangkok

Onur Mutlu, Saugata Ghose, Juan Gomez-Luna, and Rachata Ausavarungnirun, "A Modern Primer on Processing in Memory"

\textit{SAFARI} \hspace{1cm} \url{https://arxiv.org/pdf/2012.03112.pdf}
PIM Course (Fall 2022)

- **Fall 2022 Edition:**
  - [https://safari.ethz.ch/projects_and_seminars/fall2022/doku.php?id=processing_in_memory](https://safari.ethz.ch/projects_and_seminars/fall2022/doku.php?id=processing_in_memory)

- **Spring 2022 Edition:**
  - [https://safari.ethz.ch/projects_and_seminars/spring2022/doku.php?id=processing_in_memory](https://safari.ethz.ch/projects_and_seminars/spring2022/doku.php?id=processing_in_memory)

- **Youtube Livestream (Fall 2022):**
  - [https://www.youtube.com/watch?v=QLL0wQ9I4Dw&list=PL5Q2soXY2Zi8KzG2CQYRNQ0VD0G0BnK](https://www.youtube.com/watch?v=QLL0wQ9I4Dw&list=PL5Q2soXY2Zi8KzG2CQYRNQ0VD0G0BnK)

- **Youtube Livestream (Spring 2022):**
  - [https://www.youtube.com/watch?v=9e4Chnwdovo&list=PL5Q2soXY2Zi-841fUYUK9EsXKhQKRPyX](https://www.youtube.com/watch?v=9e4Chnwdovo&list=PL5Q2soXY2Zi-841fUYUK9EsXKhQKRPyX)

- **Project course**
  - Taken by Bachelor’s/Master’s students
  - Processing-in-Memory lectures
  - Hands-on research exploration
  - Many research readings

- [https://www.youtube.com/onurmutlulectures](https://www.youtube.com/onurmutlulectures)
Many questions ... How do we design the:

- compute-capable memory & controllers?
- processors & communication units?
- software & hardware interfaces?
- system software, compilers, languages?
- algorithms & theoretical foundations?
Backup Slides:
Inside A DRAM Chip
DRAM Module and Chip
Goals in DRAM Design

• Cost
• Latency
• Bandwidth
• Parallelism
• Power
• Energy
• Reliability
• Security
• ...

...
DRAM Chip
Sense Amplifier

Inverter

enable

top

bottom

Inverter
Sense Amplifier – Two Stable States

Logical “1”

Logical “0”
Sense Amplifier Operation

$V_{DD}$

$V_B$

$V_T > V_B$
DRAM Cell – Capacitor

Empty State
Logical “0”

Fully Charged State
Logical “1”

1. Small – Cannot drive circuits
2. Reading destroys the state
Capacitor to Sense Amplifier
DRAM Cell Operation

\[ \frac{1}{2} V_{DD} + \delta \]

\[ \frac{1}{2} V_{DD} \]
DRAM Subarray – Building Block for DRAM Chip

- **Row Decoder**
- **Cell Array**
- **Array of Sense Amplifiers (Row Buffer) 8Kb**
- **Cell Array**
DRAM Bank

Address

Row Decoder

Cell Array

Array of Sense Amplifiers (8Kb)

Cell Array

Array of Sense Amplifiers

Cell Array

Bank I/O (64b)

Data
DRAM Chip

Shared internal bus

Memory channel - 8bits
DRAM Operation

1. ACTIVATE Row
2. READ/WRITE Column
3. PRECHARGE

Row Address

Row Decoder

Array of Sense Amplifiers

Cell Array

Bank I/O

Column Address

Data
More on DRAM Operation: Section 2


See Section 2 for comprehensive DRAM Background

In-DRAM Bulk Bitwise Execution Engine

Vivek Seshadri
Microsoft Research India
visesha@microsoft.com

Onur Mutlu
ETH Zürich
onur.mutlu@inf.ethz.ch