CS / EE 320
Computer Organization and Assembly Language
Spring 2023
Lecture 21

Shahid Masud

Topics: Memory Hierarchy, Memory Cell Technology, Memory Array Architecture

Topics

- Examples of memory organization in memory array chip architecture using row and column decoders
- Levels of memory w.r.t. different technology, capacity, performance
- Concept of Lines in Cache and Blocks in main memory to capitalize on temporal and spatial locality
- Memory Hierarchy and CPU Connection

- **QUIZ 4 TODAY**

Memory Hierarchy

Computer Memory Combination

- Registers
  - In CPU
- Internal or Main memory
  - May include one or more levels of cache
  - “RAM”
- External memory
  - Backing store

Ideal Memory Requirement

- We want our memory to be **big** and **fast**
  - ISA promises **big**: 2^{32} memory address (4GB)
  - Want it to be **fast** because 33% of instructions are loads/stores and 100% of instructions load instructions
- But what do we have to work with?
  - Nothing that is **both big** and **fast**!

Disks are big, but super slow

SRAM is fast, but small

	Capacity	Latency	Throughput	Cost
Disk	3TB	8 ms	200 MB/s	$0.07/GB
Flash	256GB	85 µs	500 MB/s	$1.48/GB
DRAM	16GB	65 ns	10,240 MB/s	$12.50/GB
SRAM	8MB	13 ns	26,624 MB/s	$7,200/GB
SRAM	32kB	1.3 ns	47,104 MB/s	

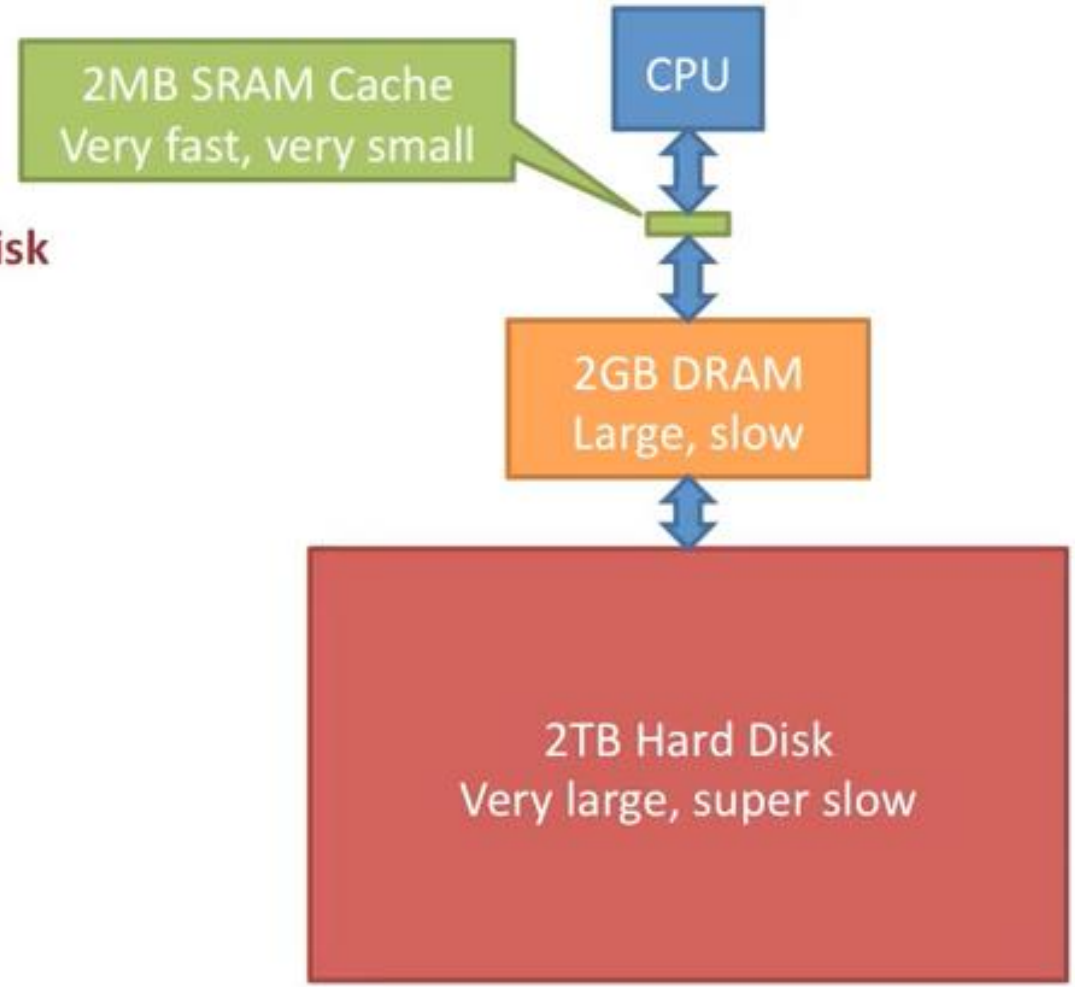
Why do we need different types of Memory?

- What do we have?
  - Hard disk: **Huge** (1000 GB) — **Super slow** (1M cycles)
  - Flash: **Big** (100 GB) — **Very slow** (1k cycles)
  - DRAM: **Medium** (10 GB) — **Slow** (100 cycles)
  - SRAM: **Small** (10 MB) — **Fast** (1-10 cycles)
- Need **fast and big**
  - Can't use **just SRAM** (too **small**)
  - Can't use **just DRAM** (too **slow** and **small**)
  - Can't use **just Flash/Hard disk** (way too **slow**)
- But we can **combine** them to get:
  - **Speed** from (small) **SRAMs**
  - **Size** from (big) **DRAM** and **Hard disk**

We'll build a hierarchy using different technologies to get the best of all of them.

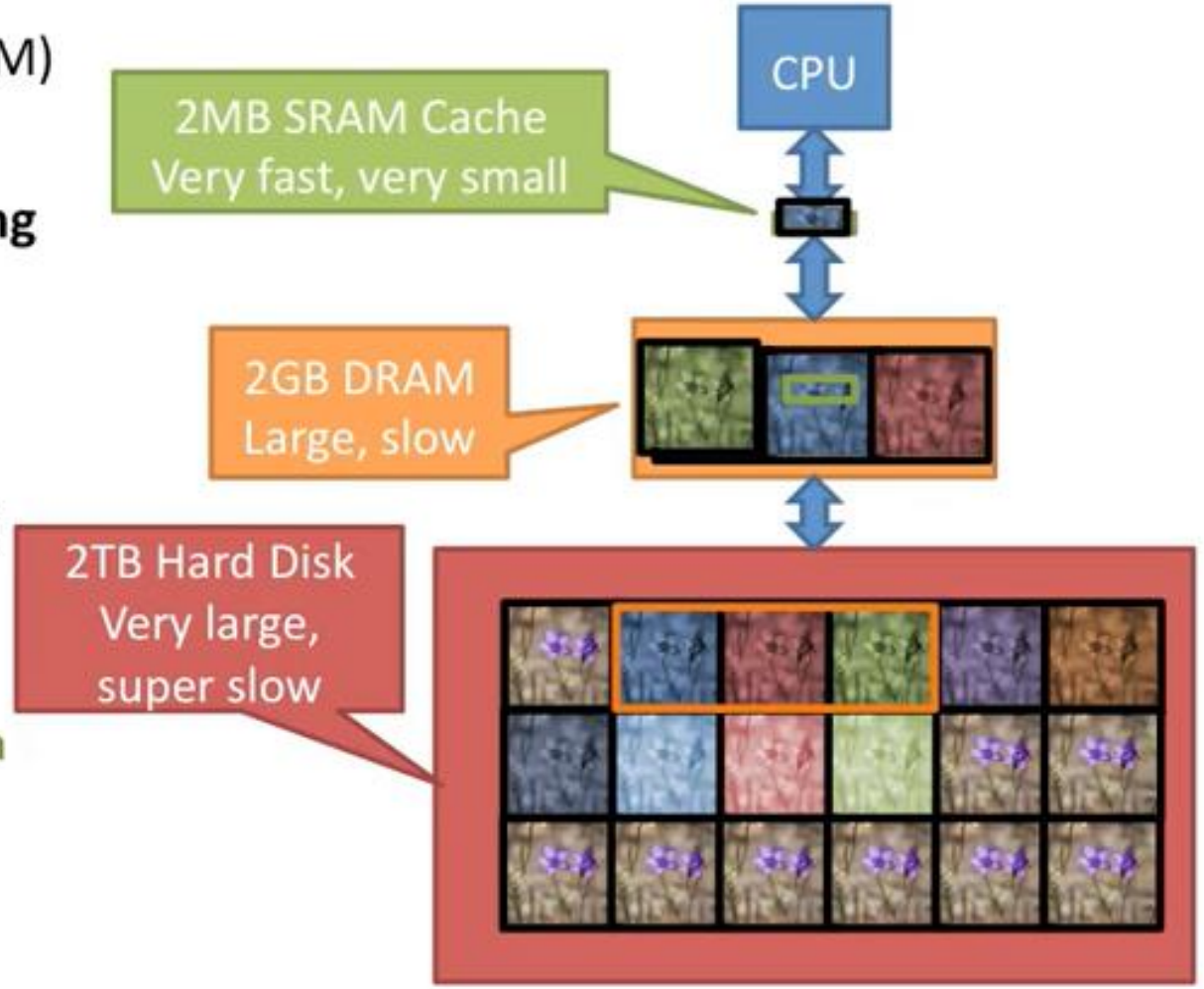
The Memory Hierarchy

- **Use:**
  - Small amounts of fast SRAM
  - Lots of slow DRAM
  - Huge amounts of super slow hard disk
- **To create the illusion of:**
  - Very **large**
  - Very **fast** (on average)
- **How do we do this?**
  - Try to keep the important data in the fast memory
  - Move the unimportant data to the slow memory

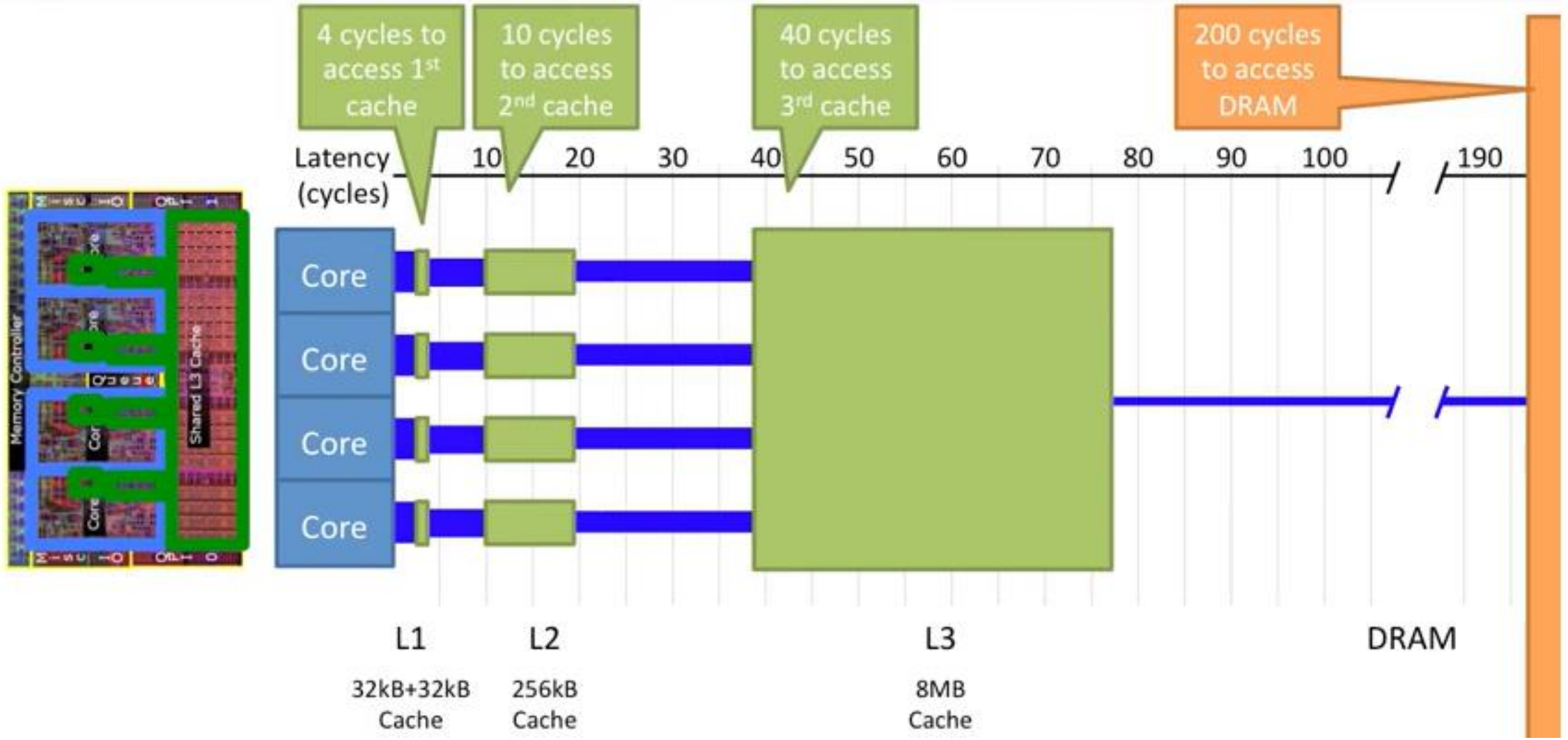


Example of Memory Hierarchy – Video Processing

- Video is large (bigger than DRAM)
- **Store on hard disk**
- **Load** just **the part we are editing** into **DRAM**
- The **CPU loads the data it is processing** into the **cache**
- Move new data into DRAM and cache as we process the video
- **Remember:**
  - Try to keep the **important data** in the **fast memory**
  - Move the **unimportant data** to the **slow memory**



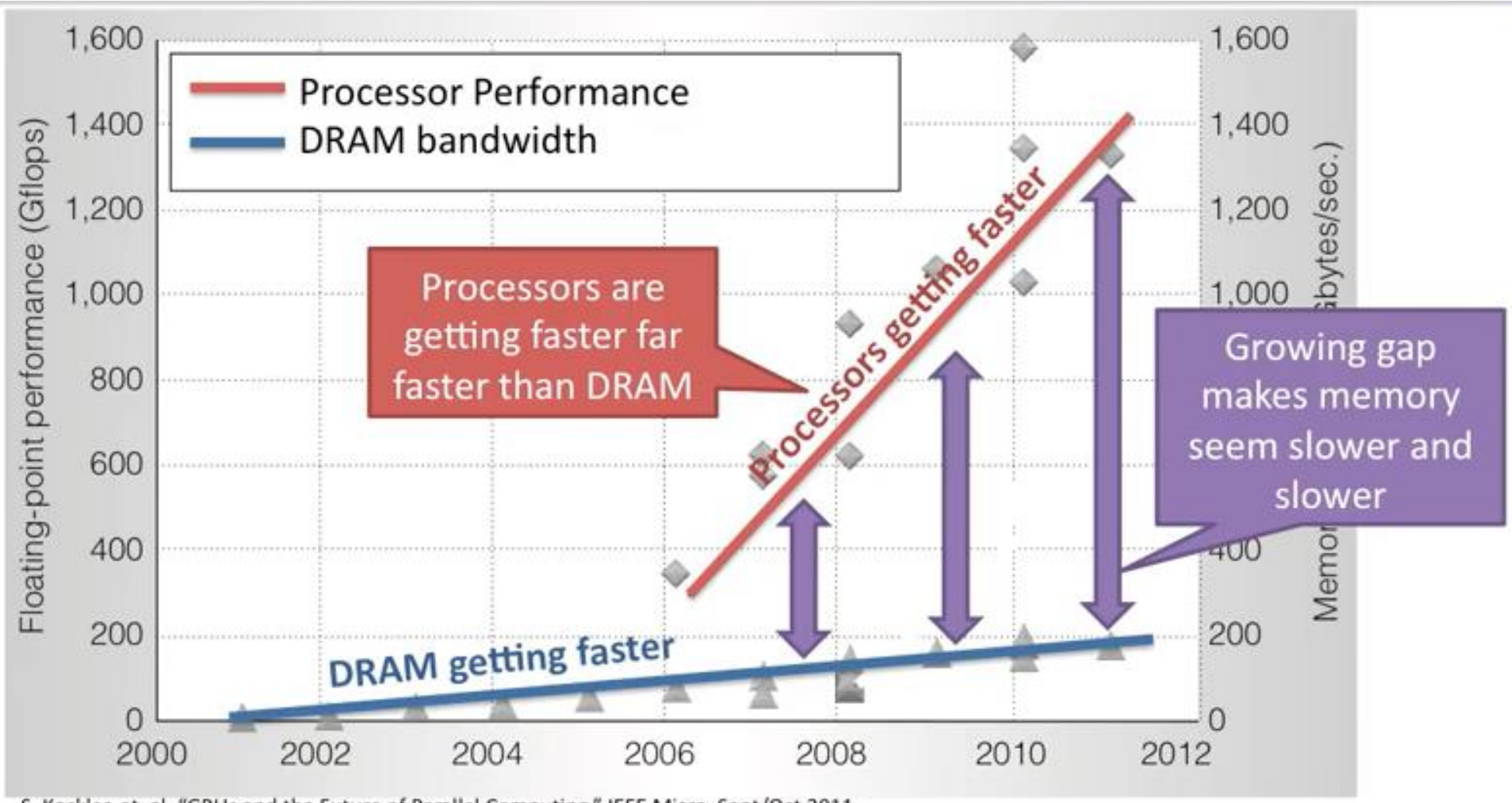
Intel Memory Hierarchy



Fast but Expensive?

- It is possible to build a computer which uses only static RAM (see later)
- This would be very fast
- This would need no cache
  - How can you cache cache?
- This would cost a very large amount

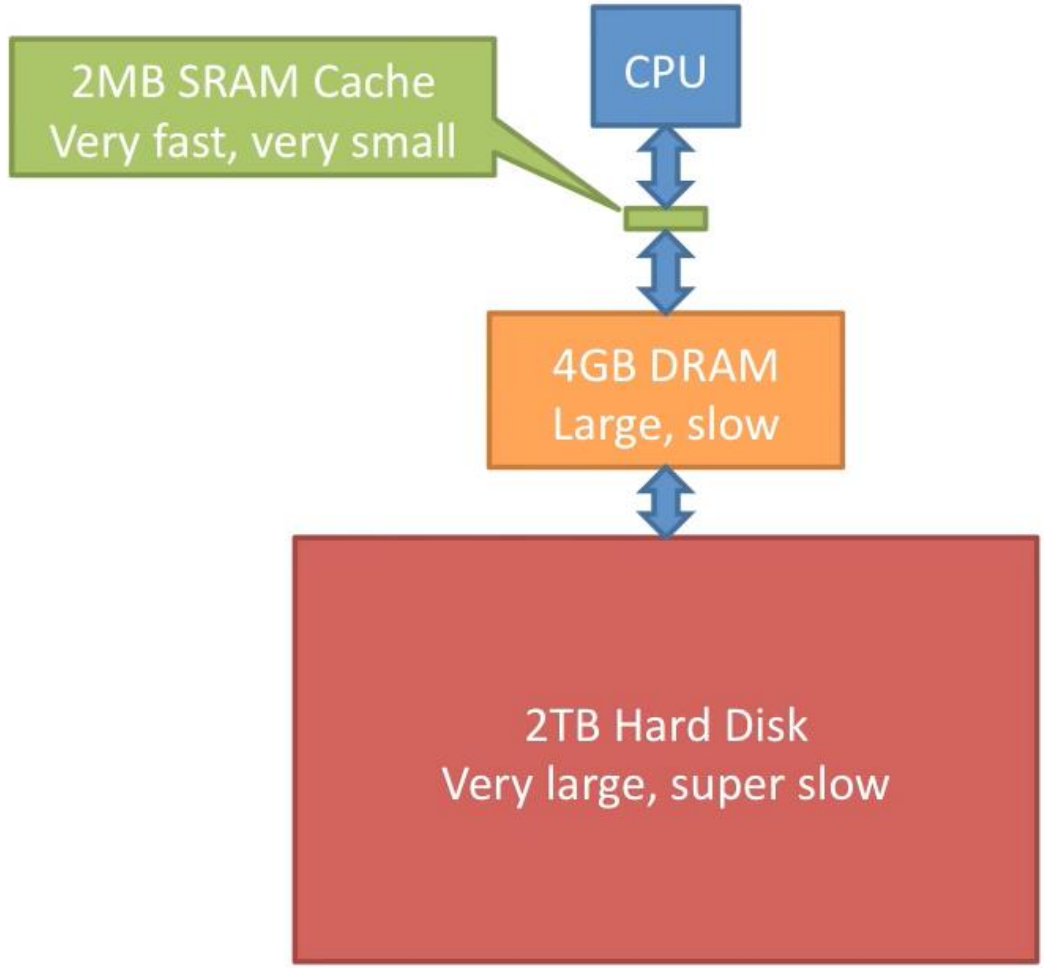
Problem with Memory and Moore's Law



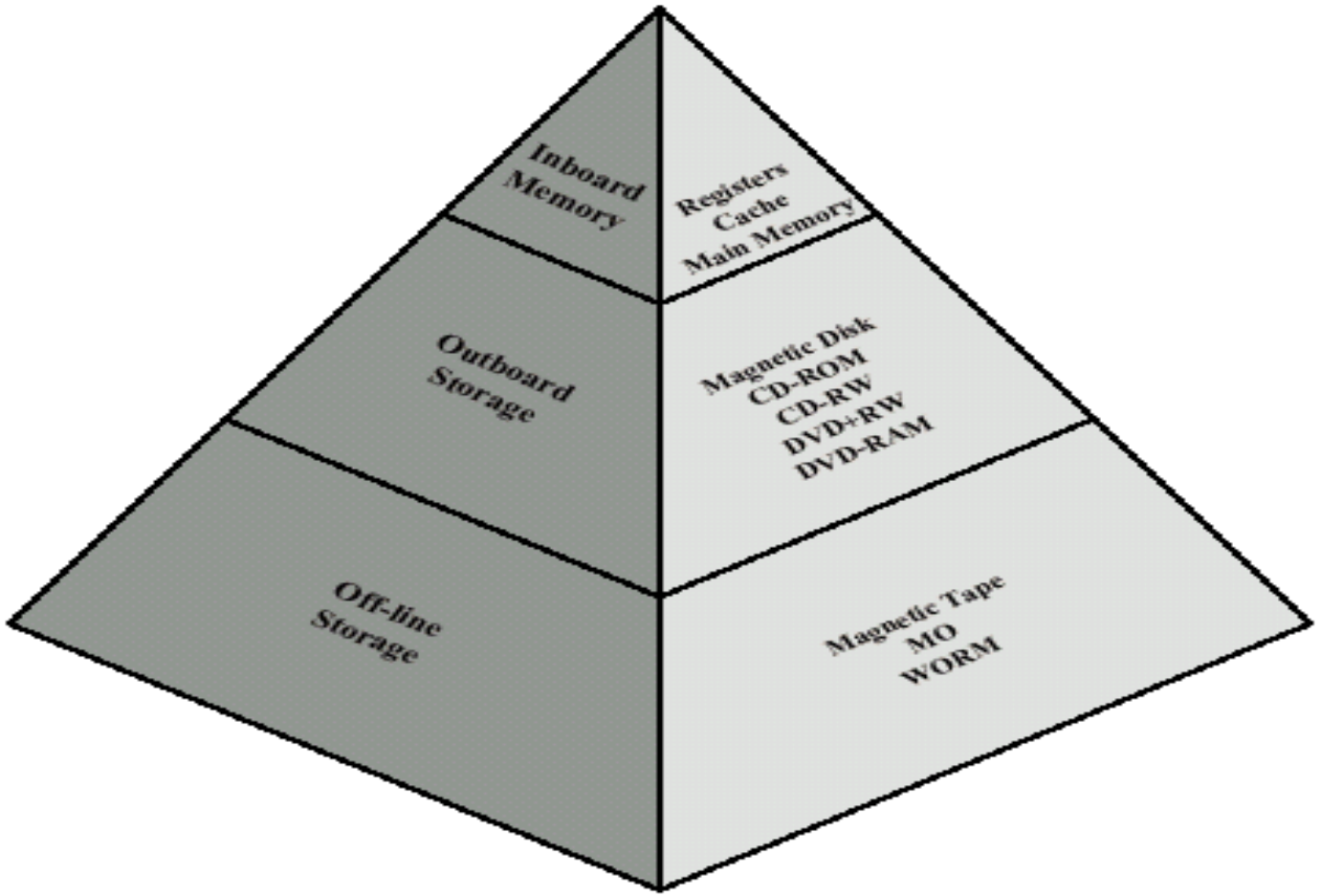
S. Keckler, et. al. "GPUs and the Future of Parallel Computing." IEEE Micro, Sept/Oct 2011.

Advantages of Memory Hierarchy

- **Very fast**
  - If we have the right data in the right place
- **Very large**
  - But possibly very slow
- **Reasonably cheap**
  - Lots of the **cheap stuff**
  - A little of the **expensive stuff**

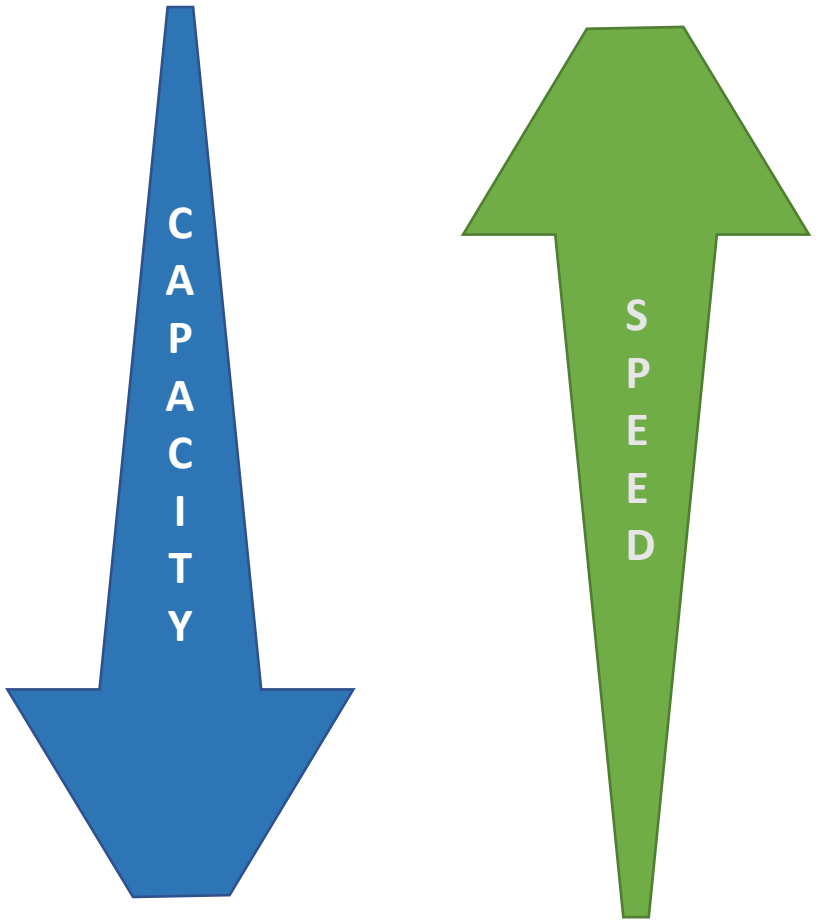


Memory Hierarchy in a Computer System



Hierarchy List

- Registers
- L1 Cache
- L2 Cache
- Main memory
- Disk cache
- Disk
- Optical
- Tape



Memory Types and Performance

Characteristics of Memory Types

- Location – CPU, Internal, External
- Capacity
- Unit of transfer
- Access method
- Performance
- Physical type
- Physical characteristics
- Organisation

Unit of Memory Transfer

- Internal
  - Usually governed by data bus width
- External
  - Usually a block which is much larger than a word
- Addressable unit
  - Smallest location which can be uniquely addressed
  - Word internally
  - Cluster on disks

Performance

- Access time
  - Time between presenting the address and getting the valid data
- Memory Cycle time
  - Time may be required for the memory to “recover” before next access
  - Cycle time is access + recovery
- Transfer Rate
  - Rate at which data can be moved

Memory Performance - Mathematically

- Access Time
  - Time between address appearing on address lines and data coming out from memory cells to data lines for RAM and vice versa.
- Memory Cycle Time
  - (Access Time + Extra time) before a second read / write can take place.
- Transfer Rate
  - For RAM $T_R = \frac{1}{Cycle\ Time}$
  - For non-RAM $T_N = T_A + \frac{N}{R}$
    - Where T_N = Avg time to read or write N bits
    - T_A = Avg Access Time
    - N = number of bits
    - R = Transfer Rate in bits / second

Physical Types of Memory

- Semiconductor
  - RAM
- Magnetic
  - Disk & Tape
- Optical
  - CD & DVD
- Physical Properties
  - Volatility
  - Eraseable
  - Power and Access

Semiconductor Memory, RAM and ROM

- RAM
  - Misnamed as all semiconductor memory is random access
  - Read/Write
  - Volatile
  - Temporary storage
  - Static or dynamic
- ROM
  - Permanent storage
  - Microprogramming (see later)
  - Library subroutines
  - Systems programs (BIOS)
  - Function tables

Static RAM

- Bits stored as on/off switches
- No charges to leak
- No refreshing needed when powered
- More complex construction
- Larger per bit
- More expensive
- Does not need refresh circuits
- Faster
- Cache

Dynamic RAM

- Bits stored as charge in capacitors
- Charges leak
- Need refreshing even when powered
- Simpler construction
- Smaller per bit
- Less expensive
- Need refresh circuits
- Slower
- Main memory

RAM Memory

Random Access Memory (RAM)

- Main memory is stored in **RAM (Random Access Memory)** .
- *Static RAM*: Implemented using a circuit similar to the D flip-flop circuit.
  - Uses 6 transistors
  - Very fast
- *Dynamic RAM*: Implemented using a transistor and a capacitor.
  - Capacitors must be refreshed periodically
  - Very high density
- RAM is *volatile* – memory cells retain their values as long as the power is on.
  - However, if the power goes off – the values disappear.
  - Registers and caches are also volatile.

ROM Memory

Read Only Memory (ROM)

- A **ROM (Read Only Memory)** is a memory where the contents of the memory are hard coded when it is manufactured.
- It is commonly used in "closed" computer systems in appliances, cars, and toys.
- In a traditional computer, the ROM is used to execute code to help boot the computer.
- ROM is *nonvolatile*– its contents remain intact even if the power is turned off.

Types of ROM

- Written during manufacture
  - Very expensive for small runs
- Programmable (once)
  - PROM
  - Needs special equipment to program
- Read “mostly”
  - Erasable Programmable (EPROM)
    - Erased by UV
  - Electrically Erasable (EEPROM)
    - Takes much longer to write than read
  - Flash memory
    - Erase whole memory electrically

PROM / FLASH Memory

The inflexibility of ROMs have given way to "programmable" ROMs or read/write nonvolatile memory:

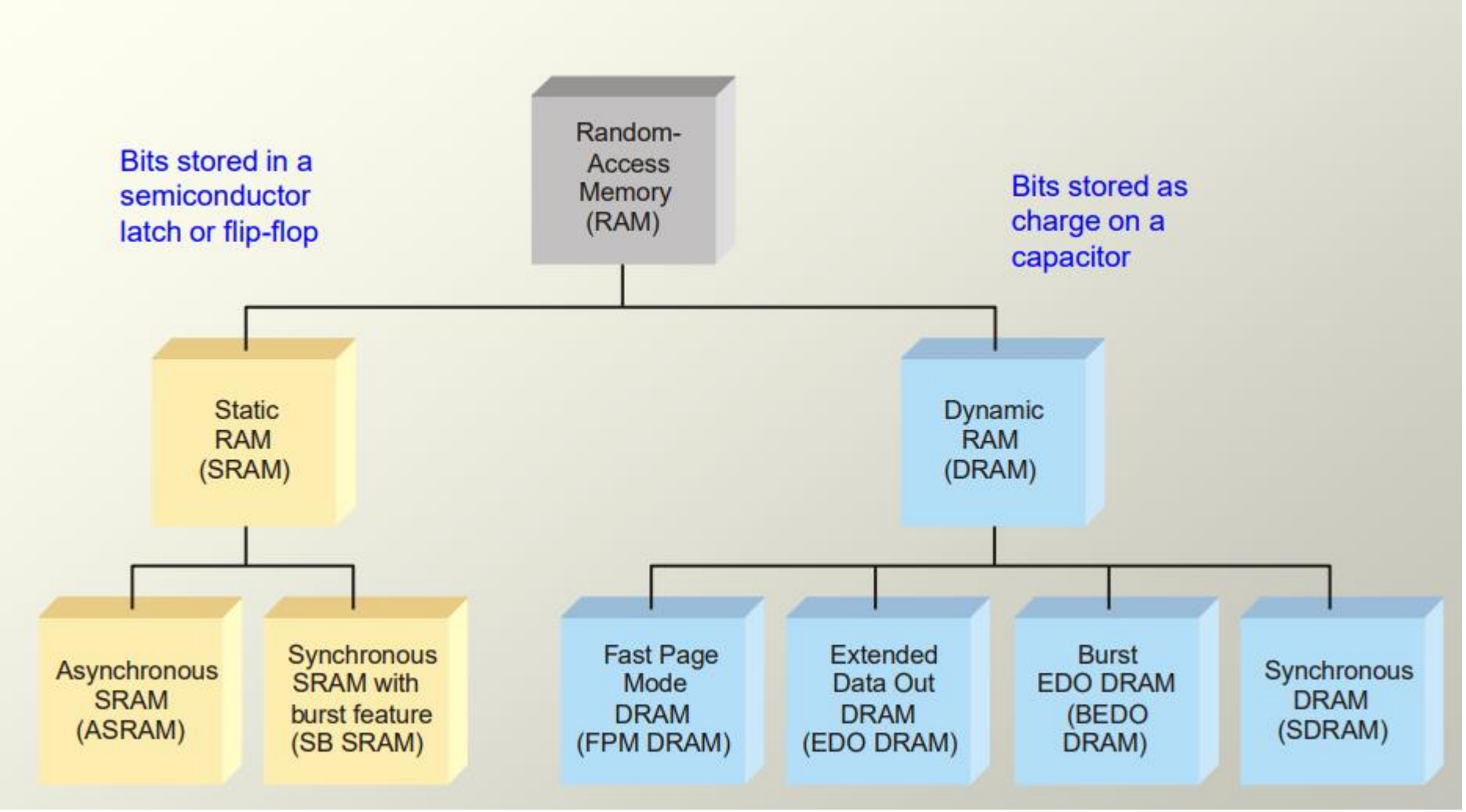
- PROM (programmable ROM): Can be programmed once.
- EPROM (erasable PROM): Can be field programmed and field erased.
- EEPROM (electrically-erasable PROM): Can be reprogrammed in place without a special device.
- Flash Memory: A form of EEPROM that is block erasable and rewritable.

SSD Memory

Solid State Disks

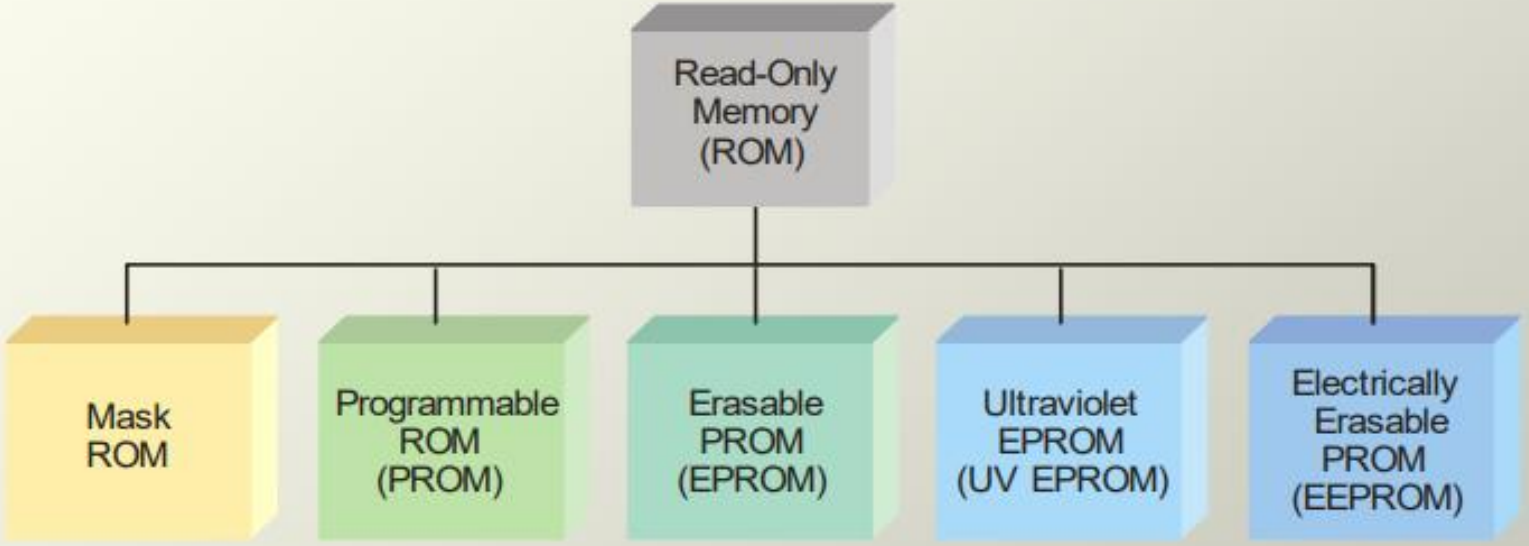
- SSDs use flash storage for random access; no moving parts.
  - Access blocks directly using block number
- Very fast reads
- Writes are slower - need a slow erase cycle (can not overwrite directly)
  - Limit on number of writes per block (over lifetime)
- Do not overwrite; garbage collect later
- Flash reads and writes faster than traditional disks
- Used in high-end I/O applications
  - Also in use for laptops, tablets

Types of RAM Memory



Types of ROM Memory

The ROM family is all considered non-volatile, because it retains data with power removed. It includes various members that can be either permanent memory or erasable.



Semiconductor Memory Types - Summary

Table 5.1 Semiconductor Memory Types

Memory Type	Category	Erasure	Write Mechanism	Volatility
Random-access memory (RAM)	Read-write memory	Electrically, byte-level	Electrically	Volatile
Read-only memory (ROM)	Read-only memory	Not possible	Masks	Nonvolatile
Programmable ROM (PROM)			Electrically	
Erasable PROM (EPROM)	Read-mostly memory	UV light, chip-level		
Electrically Erasable PROM (EEPROM)		Electrically, byte-level		
Flash memory		Electrically, block-level		

Memory Construction

Memory Cell Operation

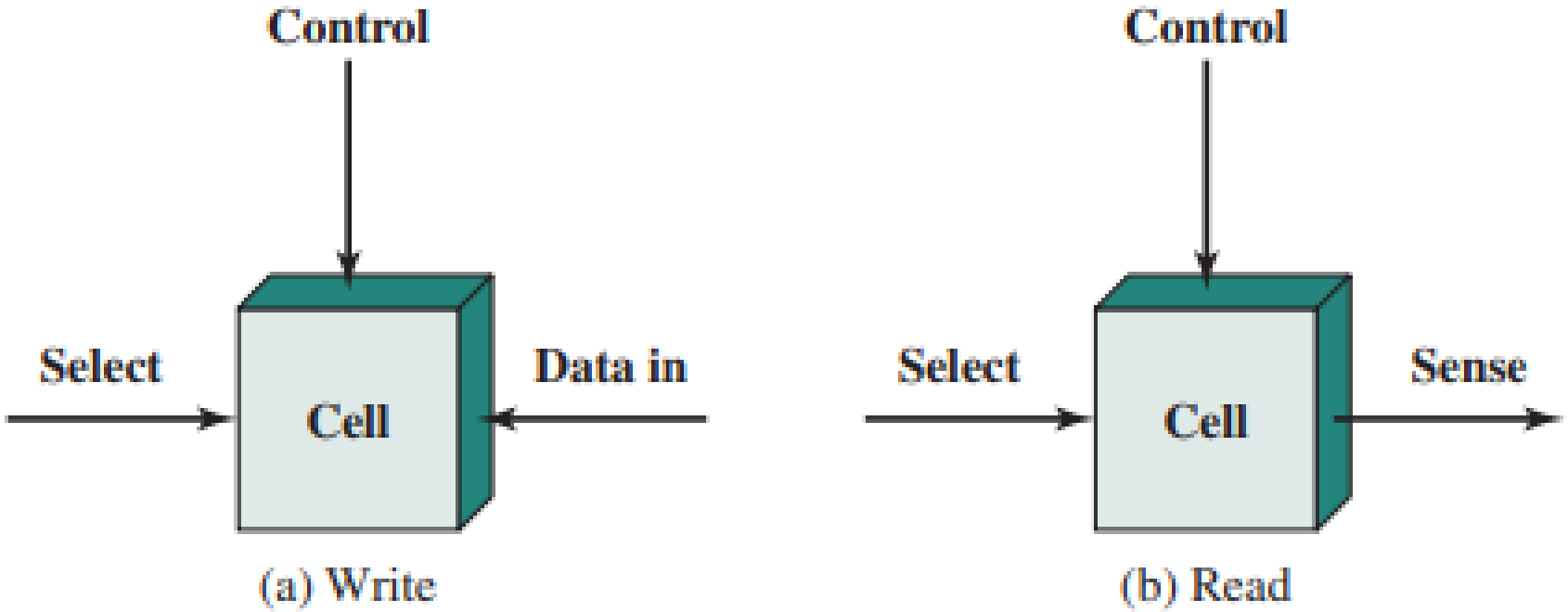
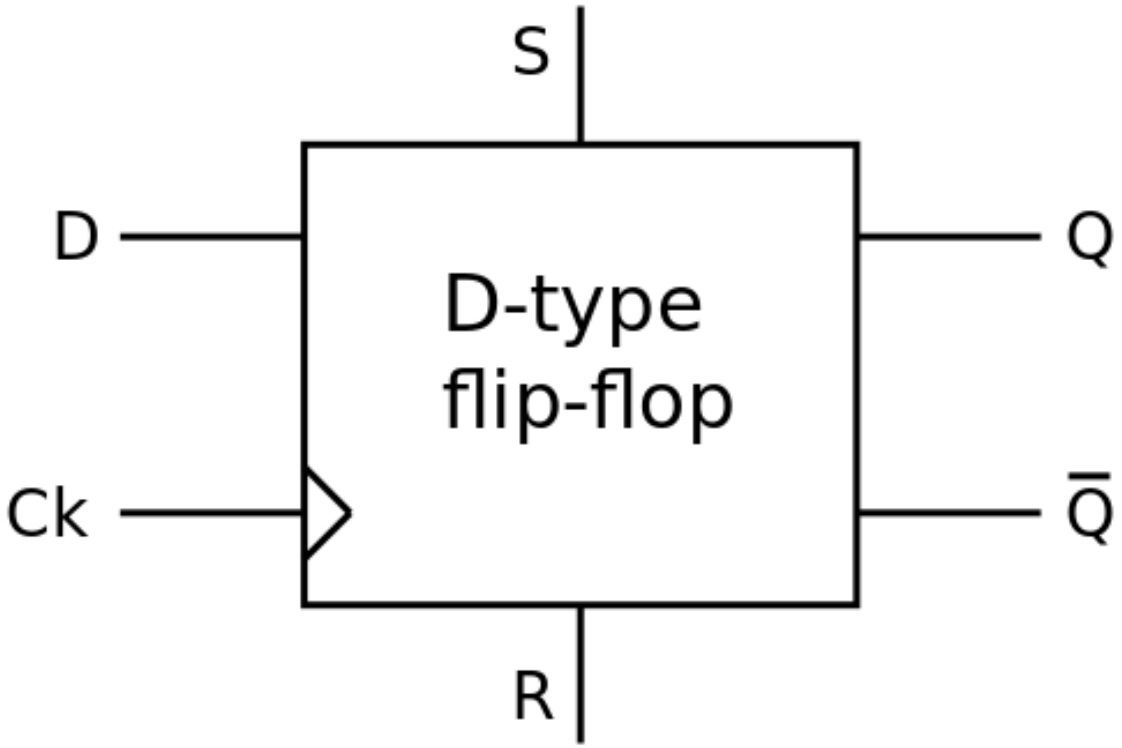


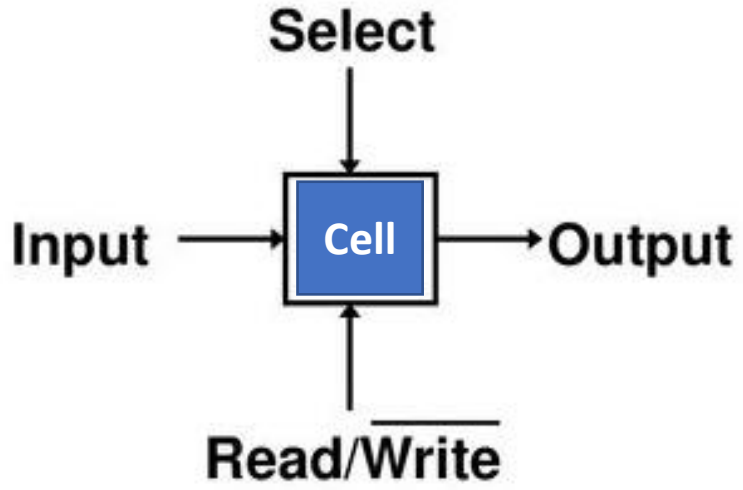
Figure 5.1 Memory Cell Operation

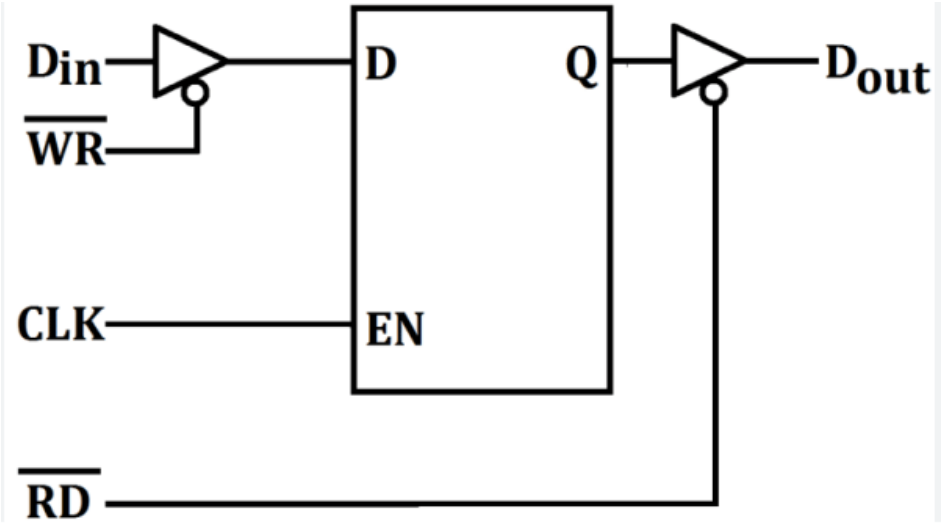
Simplest Digital Storage Element – D Flipflop



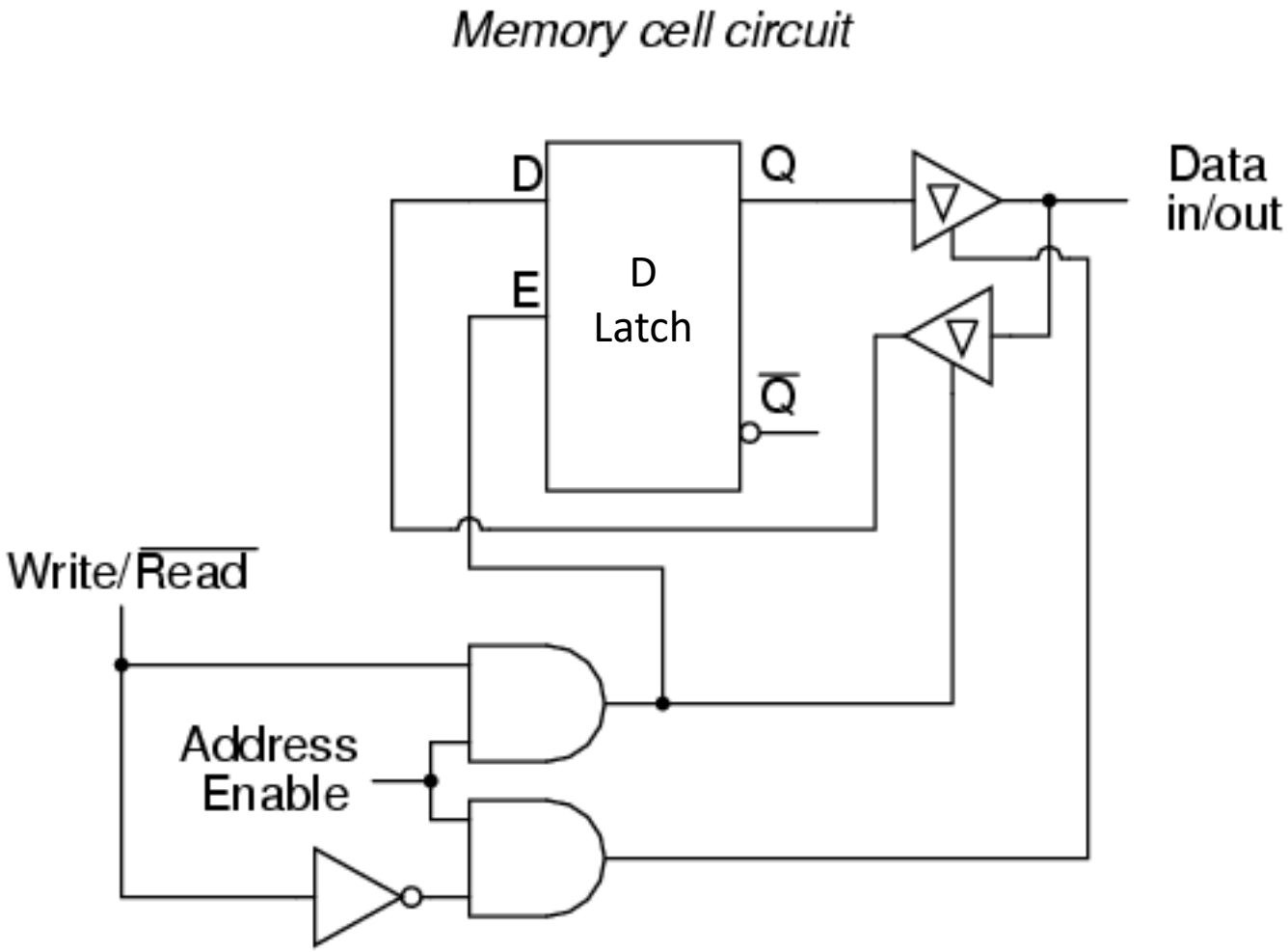
"D Latch" is a type of storage without a 'Clock'. Instead it has an 'Enable' signal.

Memory Cell Design





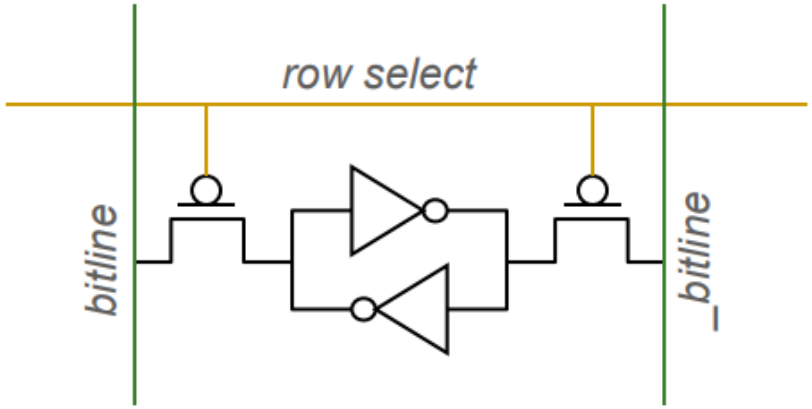
Asynchronous Memory Cell using D Latch



No Clock
Only Enable Signal

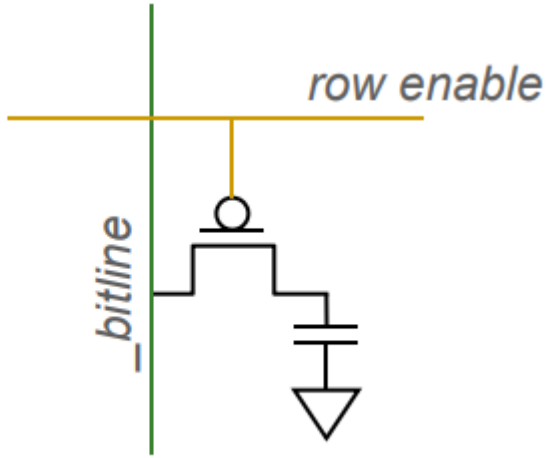
SRAM Cell Technology

- **Static random access memory**
- **Two cross coupled inverters store a single bit**
  - **Feedback path enables the stored value to persist in the "cell"**
  - **4 transistors for storage**
  - **2 transistors for access**



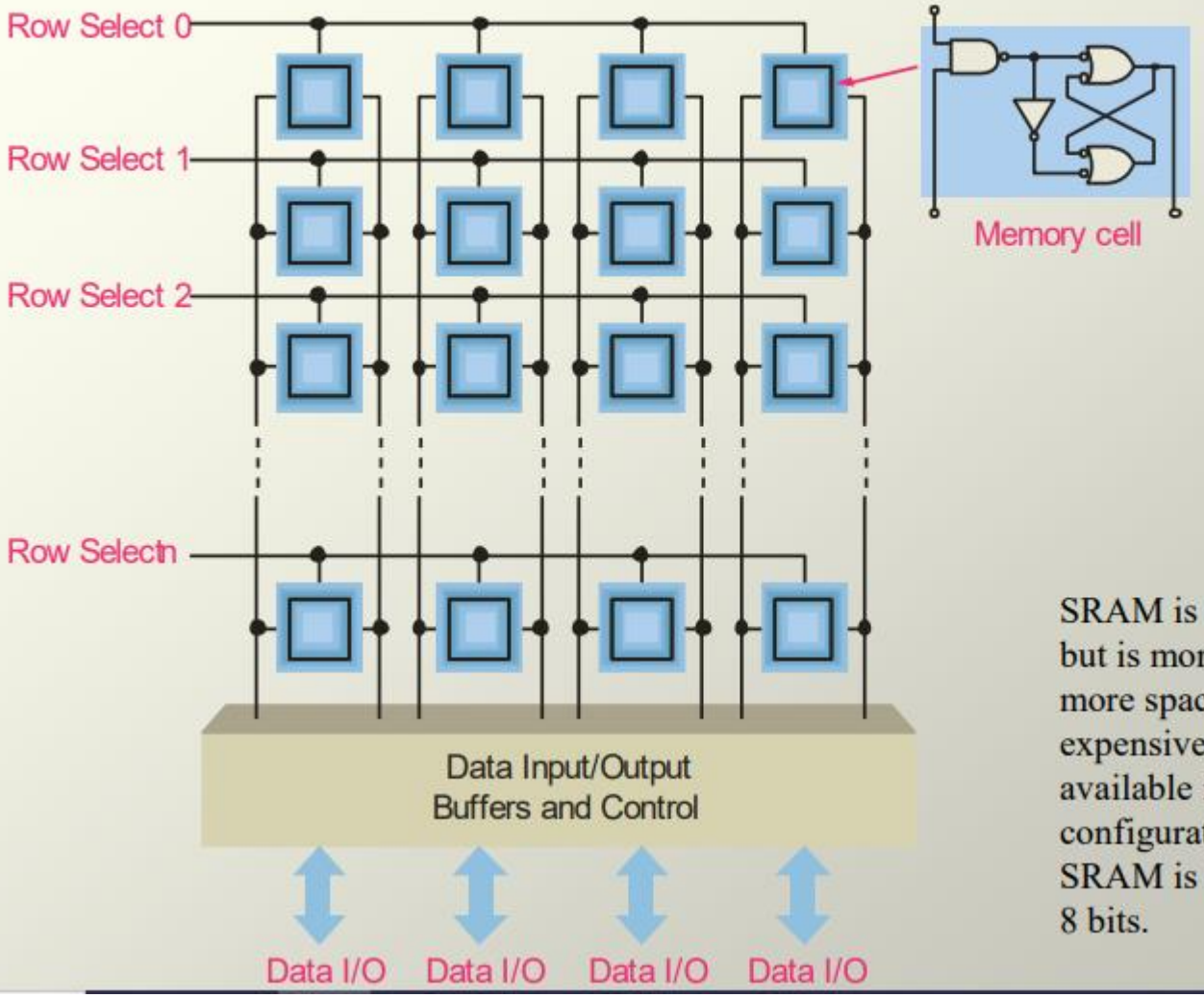
DRAM Cell Technology

- **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**



Memory Cell Organization

SRAM uses semiconductor latch memory cells. The cells are organized into an array of rows and columns.



SRAM is faster than DRAM but is more complex, takes up more space, and is more expensive. SRAMs are available in many configurations – a typical large SRAM is organized as 512 k X 8 bits.

Readings

- Chap 5 of P&H Textbook

Acknowledge: Youtube channel David Black-Schaffer for some diagrams