Memory Intro

Last updated 3/1/21
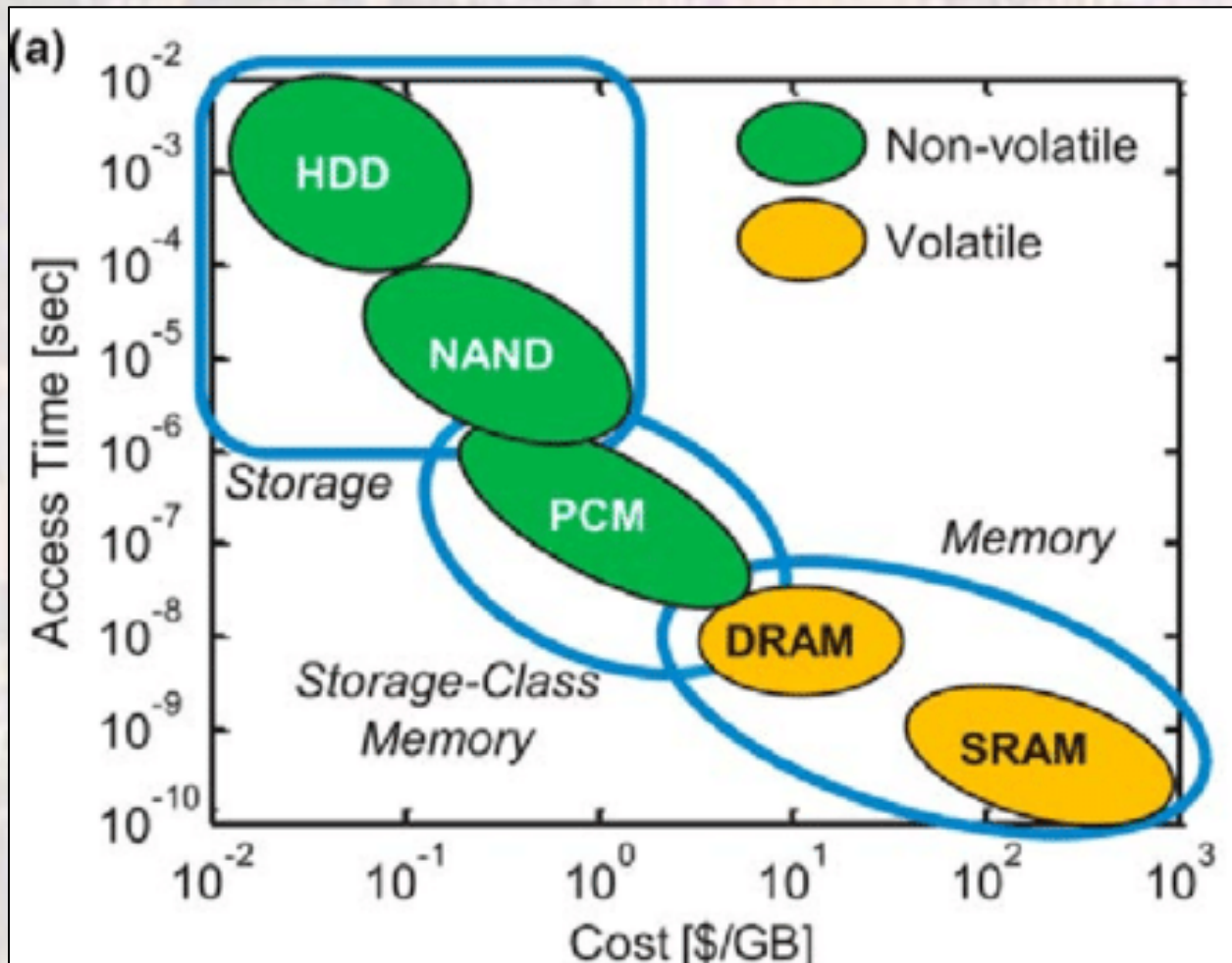
Memory Intro

- ## Memory Taxonomy

```
                              ┌──────────┐
                              │  Memory  │
                              └──────────┘
                    ┌──────────────┴───────────────────────┐
            ┌───────────────┐                    ┌───────────────────┐
            │ Random Access │                    │ Sequential Access │
            └───────────────┘                    └───────────────────┘
          ┌────────┴──────────────┐                        │
    ┌────────────┐         ┌──────────────┐              ┌──────┐
    │ Read/Write │         │  Read Only * │              │ Tape │
    └────────────┘         └──────────────┘              └──────┘
     ┌─────┴──────┐         ┌──────┴───────┐             ┌─────────┐
 ┌─────────┐  ┌────────┐ ┌──────────┐ ┌────────┐         │ Optical │
 │ Dynamic │  │ Static │ │ Mask ROM │ │ EPROM  │         └─────────┘
 └─────────┘  └────────┘ └──────────┘ └────────┘          ┌─────┐
     │        ┌────┴─────┐     │           │              │ CCD │
 ┌────────┐ ┌────────┐┌──────────┐┌──────┐ ┌────────┐     └─────┘
 │  DRAM  │ │ Power  ││ No Power ││ PROM │ │ EEPROM │
 └────────┘ │Required││ Required │└──────┘ └────────┘
            └────────┘└──────────┘
                │          │
            ┌──────┐   ┌───────┐      ┌───────┐
            │ SRAM │   │ Flash │      │ Flash │
            └──────┘   └───────┘      └───────┘
                       ┌───────┐
                       │ MRAM  │
                       └───────┘
                       ┌───────┐
                       │ FERAM │
                       └───────┘
```

* Write once or seldom
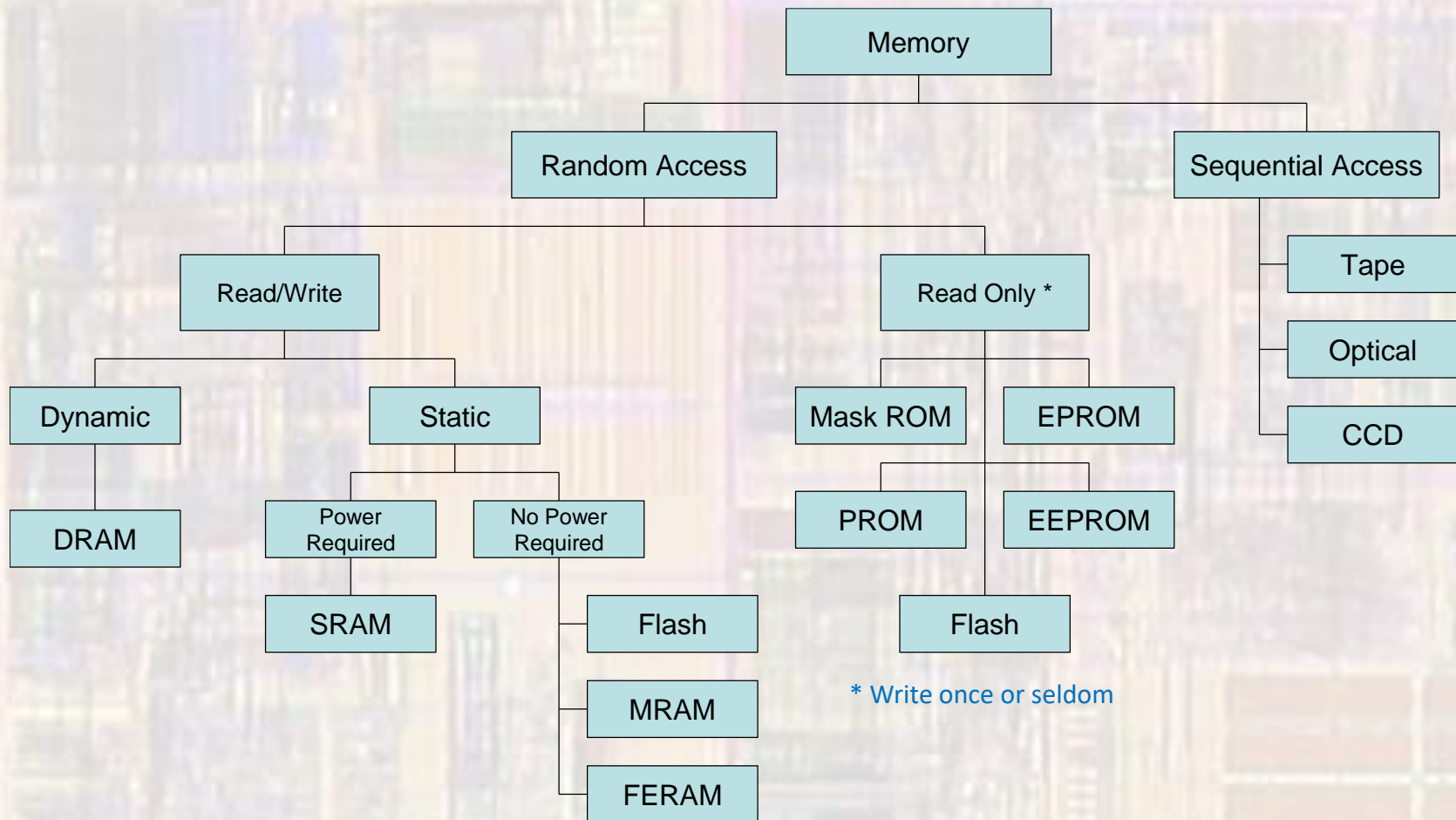
Memory Intro

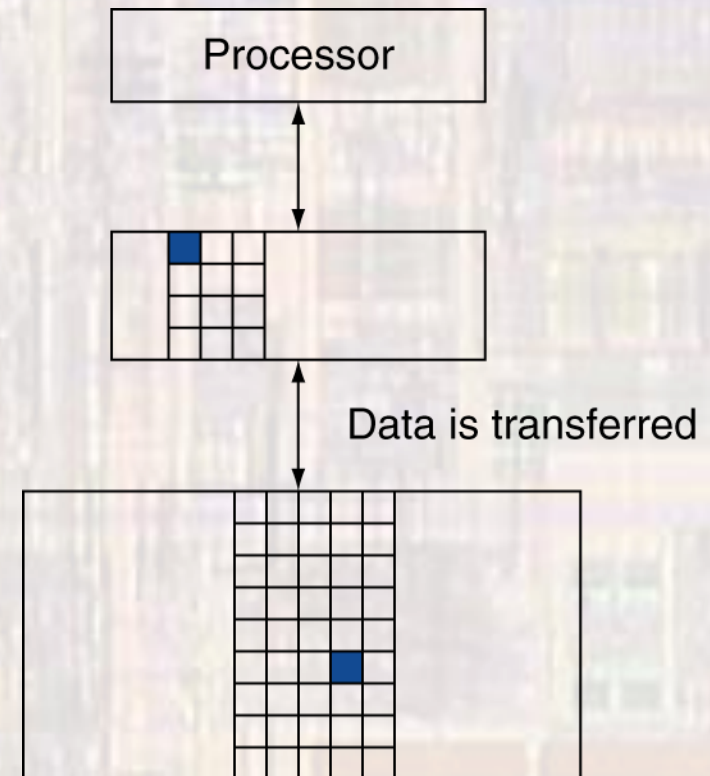- Cost/Density Comparison

Memory Intro

- Critical Memory Considerations

  - Largest – slowest
  - Fastest – most expensive

  - Can't have everything in
    a single solution

  → **Memory Hierarchy**

  - Issue with a hierarchy
    - Must transfer data up and
      down the hierarchy
   - As slow as the slowest
      level addressed

Processor

Data is transferred

Memory Intro

- Memory Hierarchy Considerations

  - Two aspects of processor applications make a memory hierarchy workable

    - Temporal Locality
      - You are more likely to uses something you recently used
        - Loops, calculated values, …

    - Spatial Locality
      - You are likely to use something that is close to something you recently used
        - Linear code, small loops, data structures

Memory Intro

- Memory Hierarchy Considerations

  - Transitions are limited between adjacent levels in the hierarchy

  - Transfer units of information
    - Line or Block
    - Different for each level

  - If what we want is in the memory we are looking at → HIT

  - If what we want is not in the memory we are looking at → MISS

Processor

Data is transferred
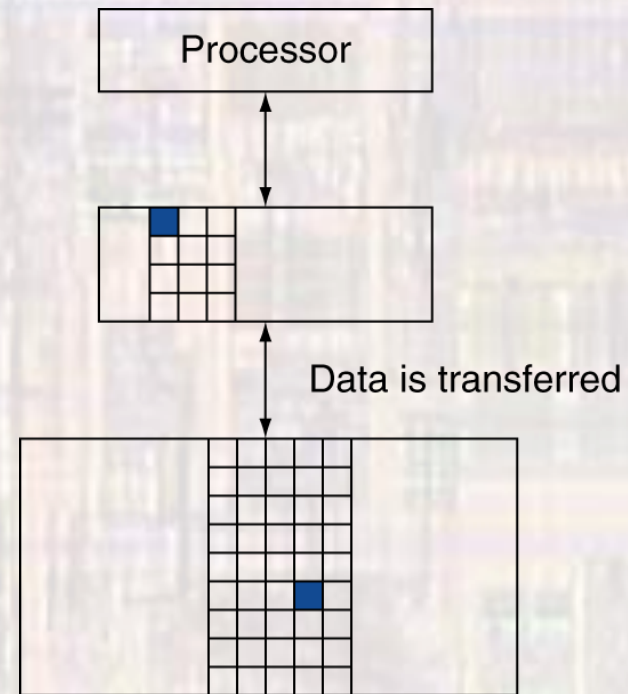
Memory Intro

- Memory Hierarchy Considerations

  - Typical System

    Registers

    Cache (SRAM)

    Main Memory (DRAM)

    Storage (HDD or Flash)

Increasing access time from CPU

CPU

L1 — SRAM, expensive Cost/mm^2
L2
L3

DRAM1 — 3D-DRAM, High B/W, Low Density

DRAM2 — High density DIMMs

SCM1 — L/S vs. R/W

SSD1/SCM2 — NAND, Low \$/mm^2

Bulk Storage — NAND, Disk, Cloud
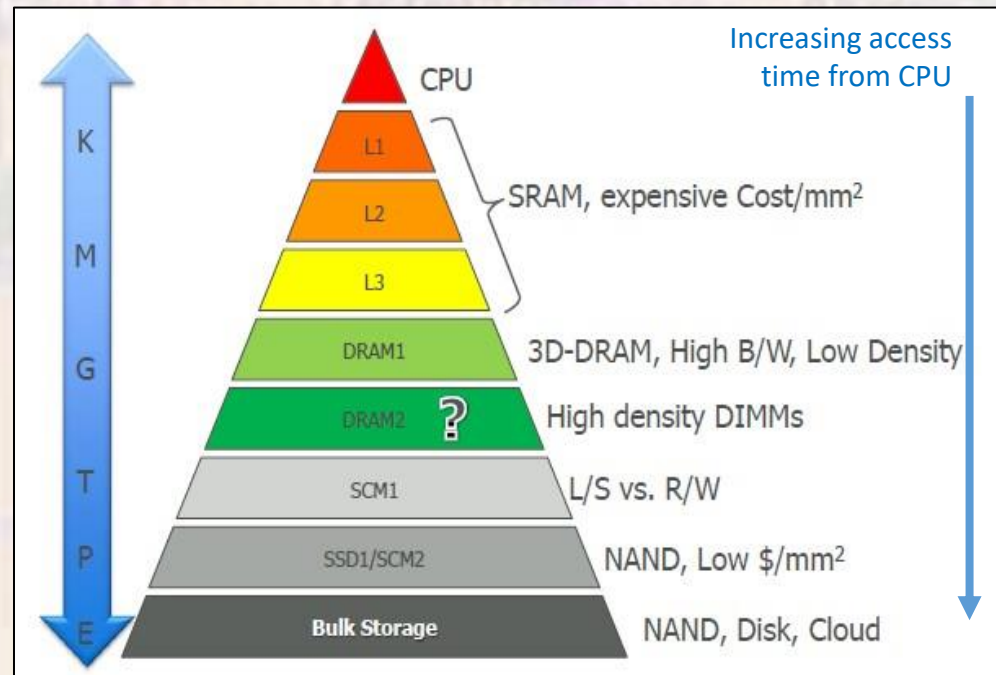
- Advanced systems may have 2,3,4 levels of cache
  - Each is progressively slower and larger
  - Size is targeted at holding entire applications
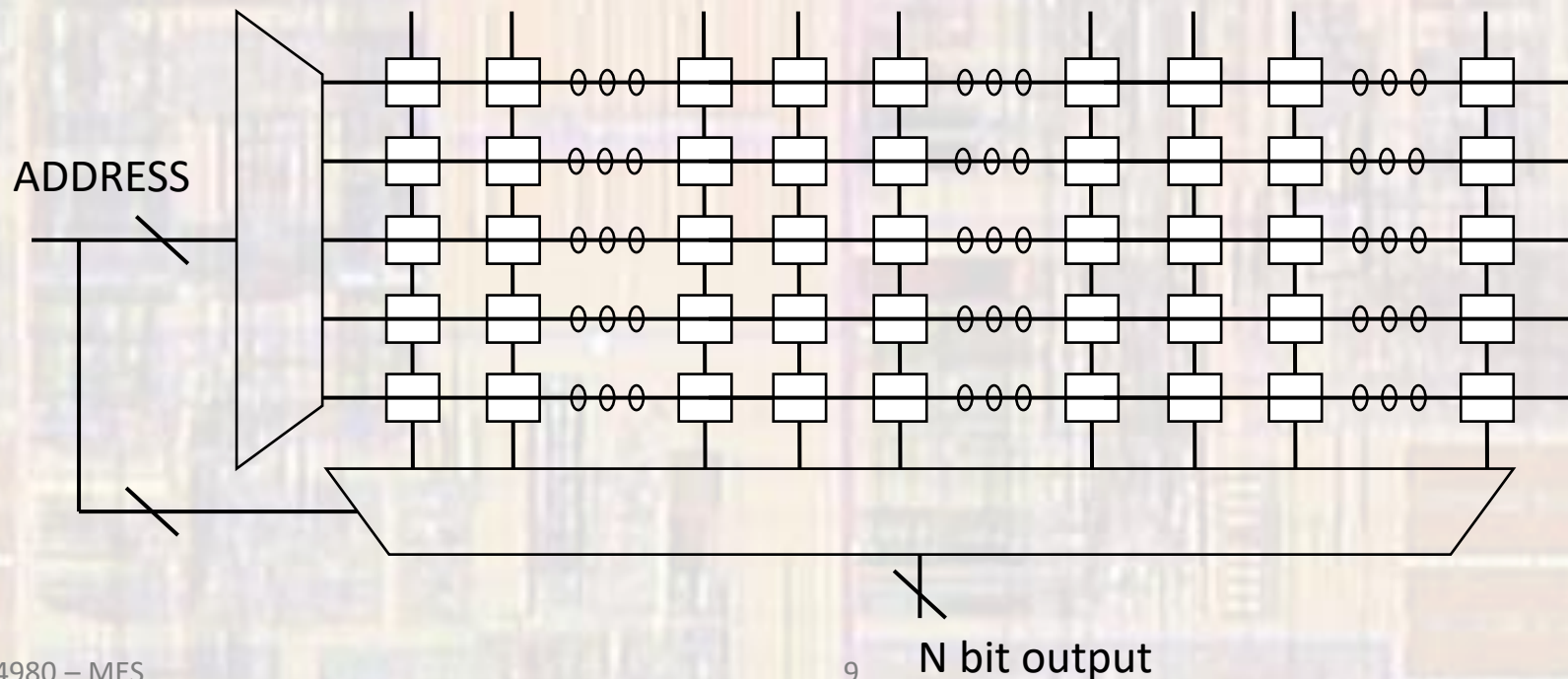
Memory Intro

- Basic Memory Topology

  - Array of single bit cells
  - Row decoder chooses 1 row

    ADDRESS

  - Rows are typically called wordlines
  - Columns are typically called bitlines

    8 bit output
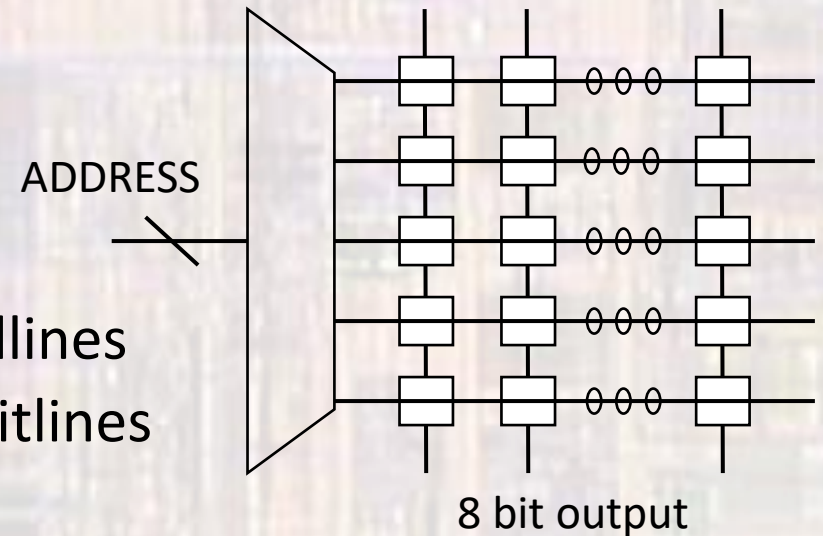
  - Non optimal
    - Physical implementation
      - Array
      - Decoder
    - Speed

Memory Intro

- General Memory Topology

  - Array of single bit cells
  - Row decoder chooses 1 row
  - Column decoder chooses one column
    - 1,4,8,16,32,64,128,... bits/column

ADDRESS

N bit output

Memory Intro

- General Memory Topology

  - Read the description very carefully

    - 1Gb memory in a x4 configuration
      - Total of 1Gb

      Is different than

    - 1Gb x 4 memory
      - Total of 4Gb

Memory Intro

- ## General Memory Topology

  - Example
  - 16Mb memory in a x4 configuration

  - x4 means each column is 4 bits
  - 16Mb → 16,777,216 bits
  - 16Mb in a x4 configuration → 4,194,304 - individual addresses
  - 4,194,304 addresses → 22 address bits

  - x4 means 4 bit cells for every column
  - Assuming a square memory array and a square bit cell → 4 times as many rows as columns

  - 22 address bits → 12 bits of row address and 10 bits of column address

Memory Intro

- General Memory Topology

16Mb in x4 configuration

ADDRESS

12

10

4 bit output