EECS151/251A
Introduction to Digital Design and ICs

Lecture 24: Other Memory

Sophia Shao

SK Hynix to Buy Intel's NAND Memory Business For $9 Billion

In a joint press release issued early this morning, SK Hynix and Intel have announced that Intel will be selling the entirety of its NAND memory business to SK Hynix. The deal, which values Intel's NAND holdings at $9 billion, will see the company transfer over the NAND business in two parts, with SK Hynix eventually acquiring all IP, facilities, and personnel related to Intel's NAND efforts. Notably, however, Intel is not selling their overarching Non-Volatile Memory Solutions Group; instead the company will be holding on to their Optane memory technology as they continue to develop and sell that technology.

https://www.anandtech.com/show/16182/sk-hynix-to-buy-intels-nand-memory-business-for-9-billion

Review

- SRAM cells sized for stability and writeability.
- Memory decoder & pre-decoder:
  - Decoder is a series of AND gates that drive word lines.
  - One decoder per read/write port.
  - Broken into predecoder for better area and delay.

- Cache
- DRAM
- Content-Addressable Memory
- Flash

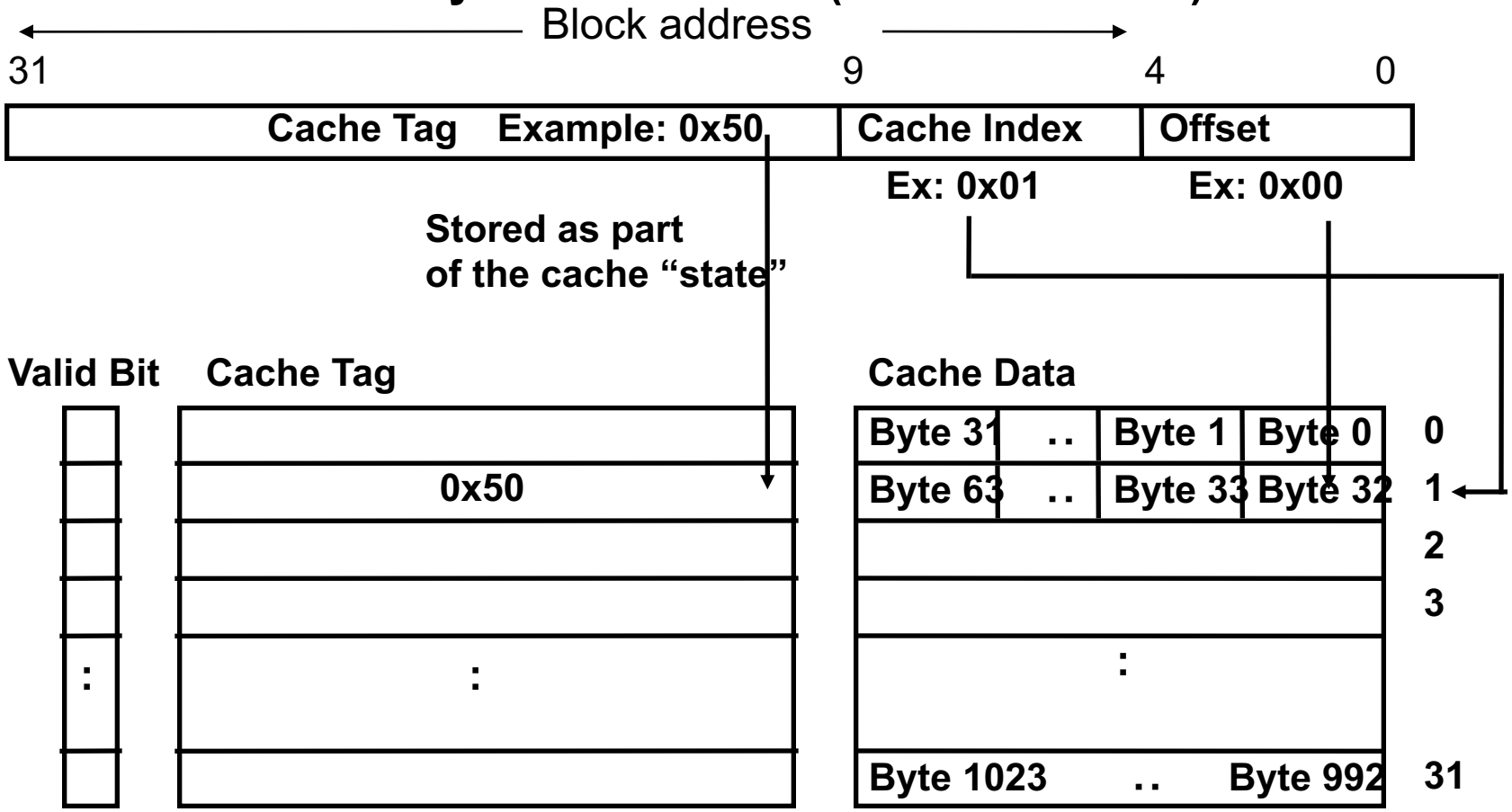
Caches (Review from 61C)

- Two Different Types of Locality:
  - Temporal locality (Locality in time): If an item is referenced, it tends to be referenced again soon.
  - Spatial locality (Locality in space): If an item is referenced, items whose addresses are close by tend to be referenced soon.
- By taking advantage of the principle of locality:
  - Present the user with as much memory as is available in the cheapest technology.
  - Provide access at the speed offered by the fastest technology.
- DRAM is slow but cheap and dense:
  - Good choice for presenting the user with a BIG memory system
- SRAM is fast but expensive and not as dense:
  - Good choice for providing the user FAST access time.

Example: 1 KB Direct Mapped Cache with 32 B Blocks

For a 2^N-byte cache:

- **The uppermost (32 - N) bits are always the Cache Tag**
- **The lowest M bits are the byte-select offset (Block Size = 2^M)**

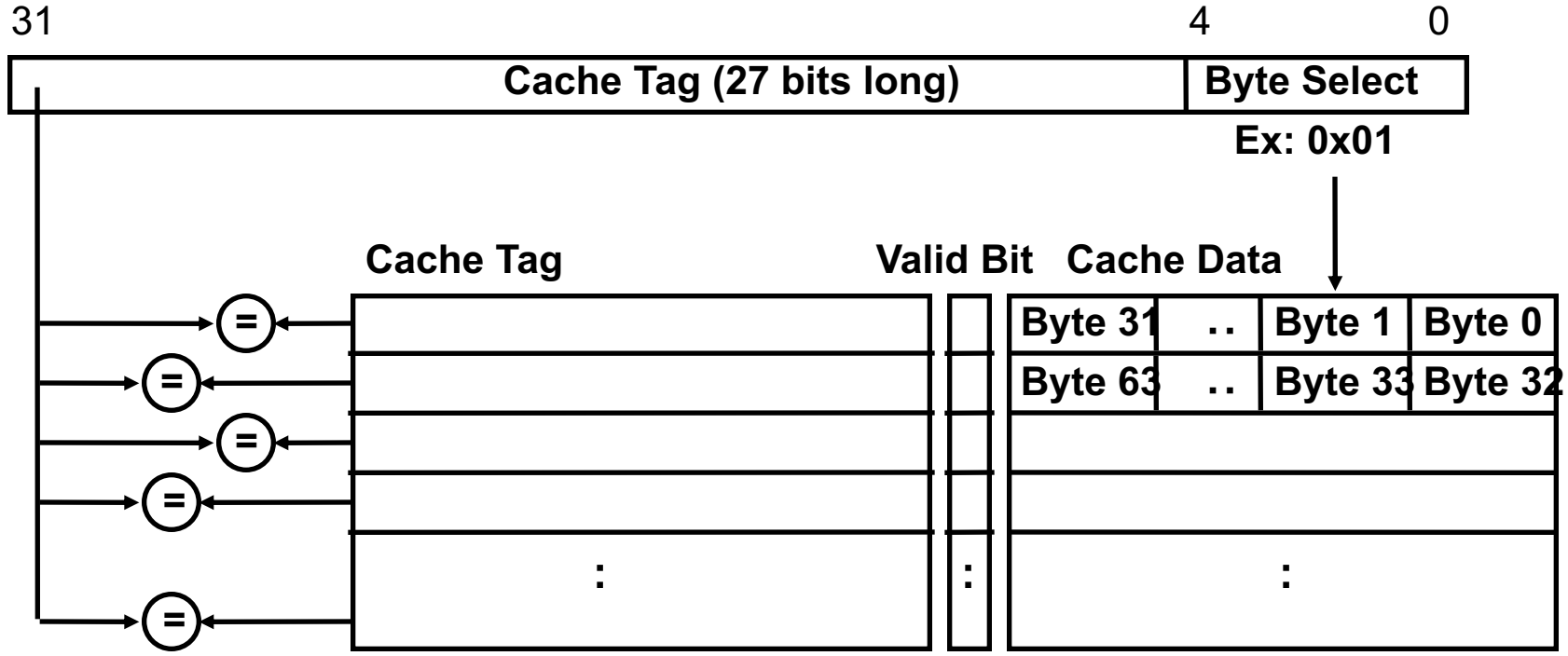


Fully Associative Cache

Fully Associative Cache

- Ignore cache Index for now
- Compare the Cache Tags of all cache entries in parallel (expensive…)_
- Example: Block Size = 32 B blocks, we need N 27-bit comparators

By definition: Conflict Miss = 0 for a fully associative cache



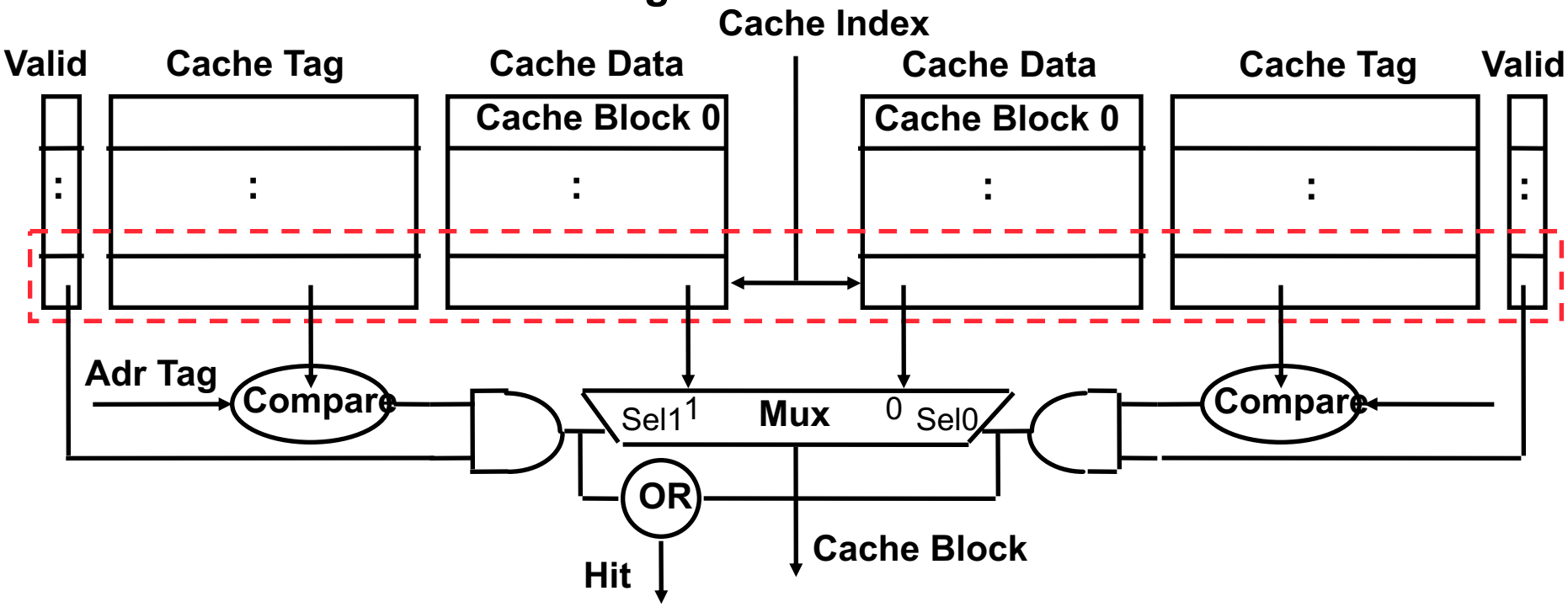
Set Associative Cache

N-way set associative: N entries for each Cache Index

- N direct mapped caches operates in parallel

Example: Two-way set associative cache

- Cache Index selects a “set” from the cache
- The two tags in the set are compared to the input in parallel
- Data is selected based on the tag result



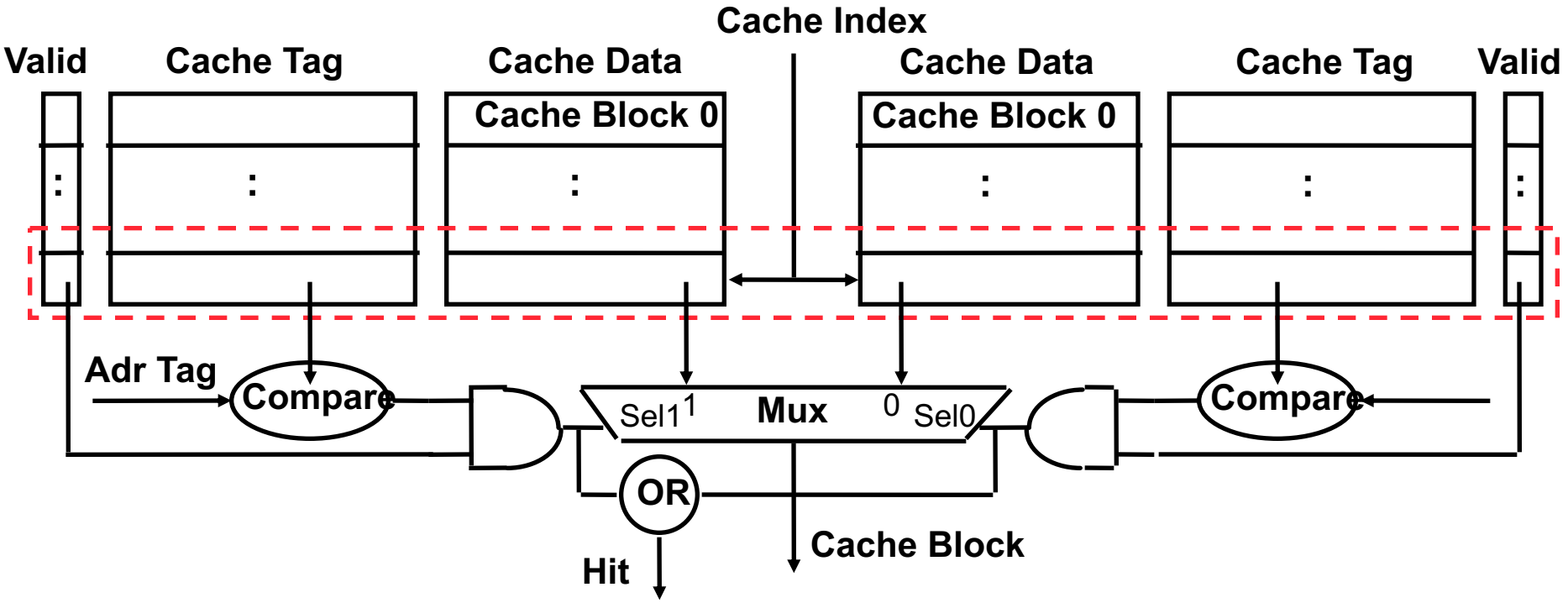
Disadvantage of Set Associative Cache

N-way Set Associative Cache versus Direct Mapped Cache:

- **N comparators vs. 1**
- **Extra MUX delay for the data**
- **Data comes AFTER Hit/Miss decision and set selection**

In a direct mapped cache, Cache Block is available BEFORE Hit/Miss:

- **Possible to assume a hit and continue. Recover later if miss.**



Block Replacement Policy

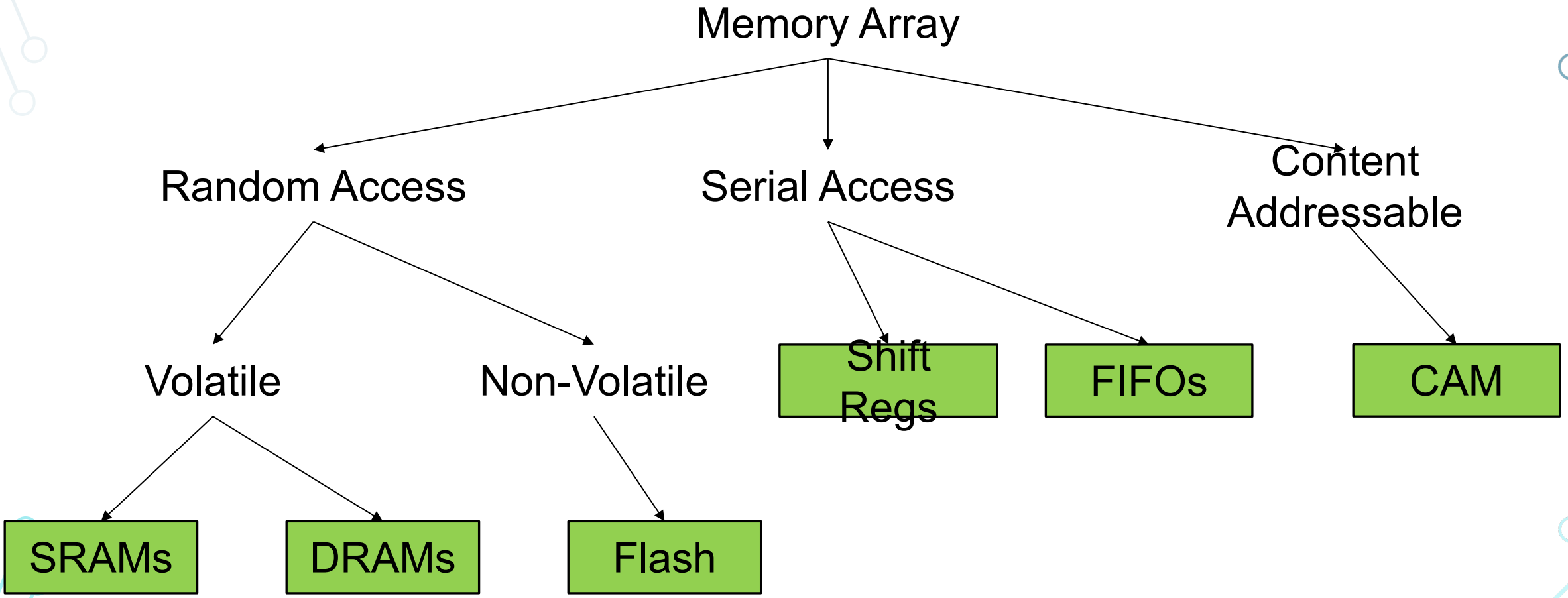
- Direct-Mapped Cache
  - index completely specifies position which position a block can go in on a miss
- N-Way Set Assoc
  - index specifies a set, but block can occupy any position within the set on a miss
- Fully Associative
  - block can be written into any position
- Question: if we have the choice, where should we write an incoming block?
  - If there's a valid bit off, write new block into first invalid.
  - If all are valid, pick a replacement policy
    - rule for which block gets "cached out" on a miss.

Block Replacement Policy: LRU

- LRU (Least Recently Used)
  - Idea: cache out block which has been accessed (read or write) least recently
  - Pro: temporal locality ➔ recent past use implies likely future use: in fact, this is a very effective policy
  - Con: with 2-way set assoc, easy to keep track (one LRU bit); with 4-way or greater, requires more complicated hardware and more time to keep track of this

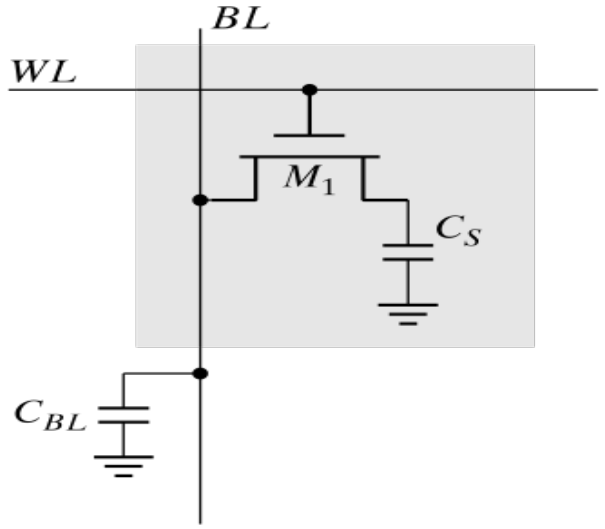
- Cache
- DRAM
- Content-Addressable Memory
- Flash

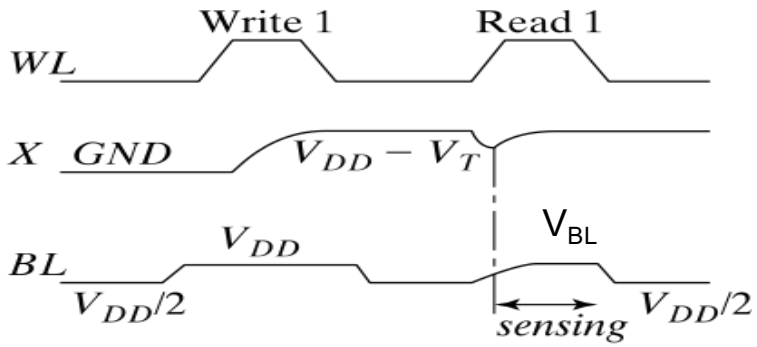
Memory Overview



1-Transistor DRAM Cell

- DRAMs store their contents as charge on a capacitor rather than in a feedback loop (as SRAM).





V_{BIT}= 0 or (V_{DD} – V_T)

Write: C s is charged or discharged by asserting WL and BL.

Read: Charge redistribution takes places between bit line and storage capacitance

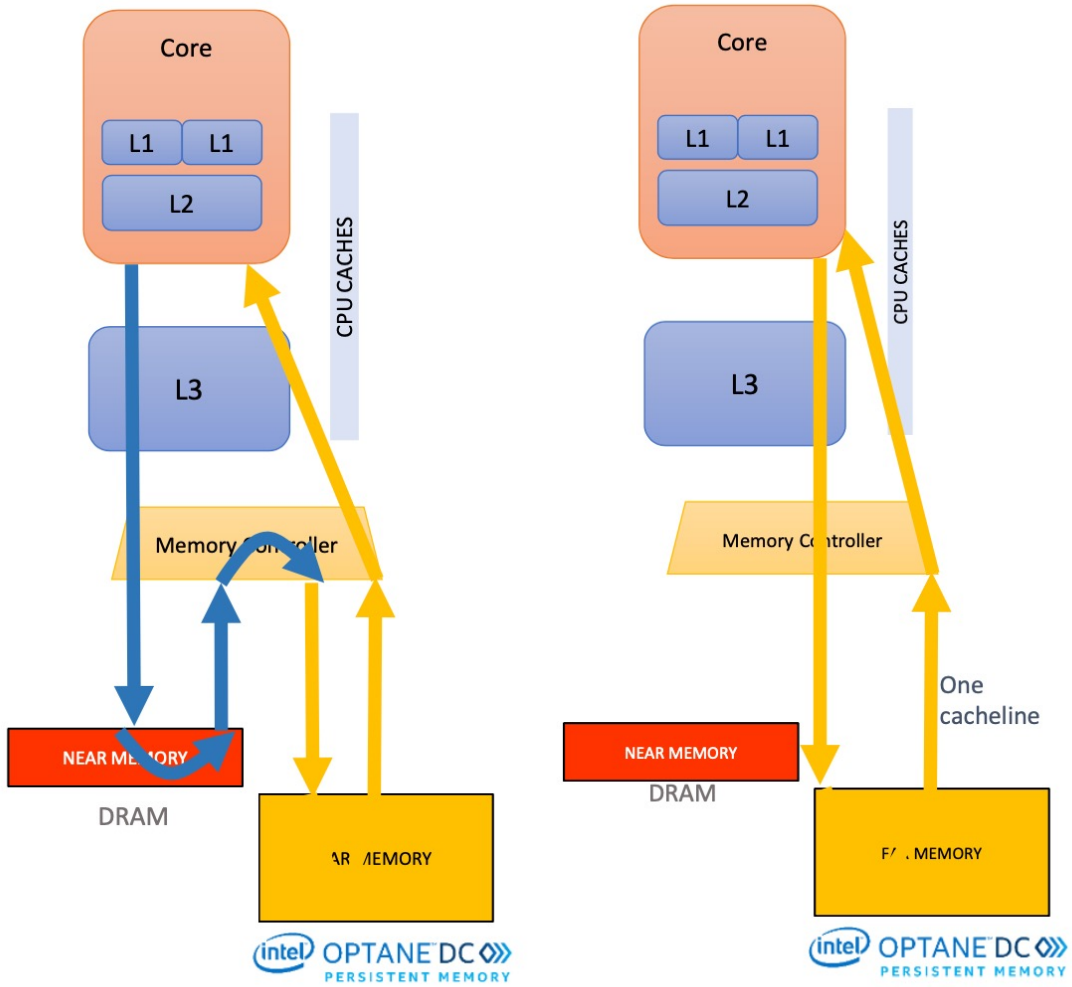
$$\Delta V = \frac{V_{DD}}{2}\frac{C_S}{C_S + C_{BL}}$$

Voltage swing is small; typically hundreds of mV.

- To get sufficiently large C_s, special IC process is used.
- Cell reading is destructive, therefore read operation always is followed by a write-back
- Cell looses charge (leaks away in ms - highly temperature dependent), therefore cells occasionally need to be “refreshed” - read/write cycle

Non-Volatile Main Memory

- Intel Optane DC Persistent Memory
- Non-Volatile
- Storage based on resistance:
  - High resistivity : 0
  - Low Resistivity: 1
- High capacity:
  - 128, 256, 512 GB
- Modes:
  - Memory Mode
  - App-directed Mode



Administrivia

- One more regular lecture: this Thursday.
  - No more lecture next Thursday.
  - Use that slot to work on your project.
- Guest lecture next Tuesday on SystemVerilog.
- HW8: due this Friday.
- HW9: Released this week. Due next Friday.
- FPGA Lab:
  - 5/2 Mon Final Project checkoff
- ASIC Lab
  - 5/6 Friday Final Presentation
- Final: 5/12 3-6pm

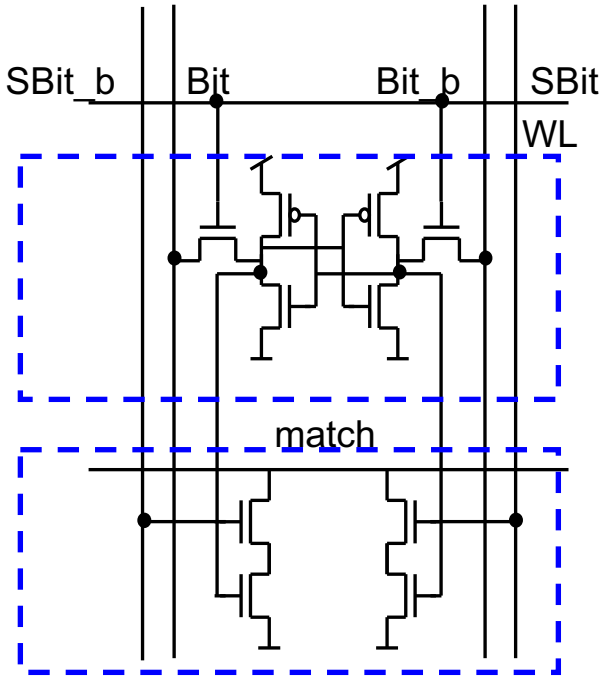
Administrivia

- Course Survey
  - https://course-evaluations.berkeley.edu/berkeley/
  - We VALUE your feedback!
  - Tell us your experience!
  - Tell us what worked and what could be improved!
  - Extra credits:
    - 1pt if you submit a confirmation screenshot (private post on Piazza)!
    - 1 more pt for everyone who complete the survey if we hit 70% response rate!

- Cache
- DRAM
- Content-Addressable Memory
- Flash

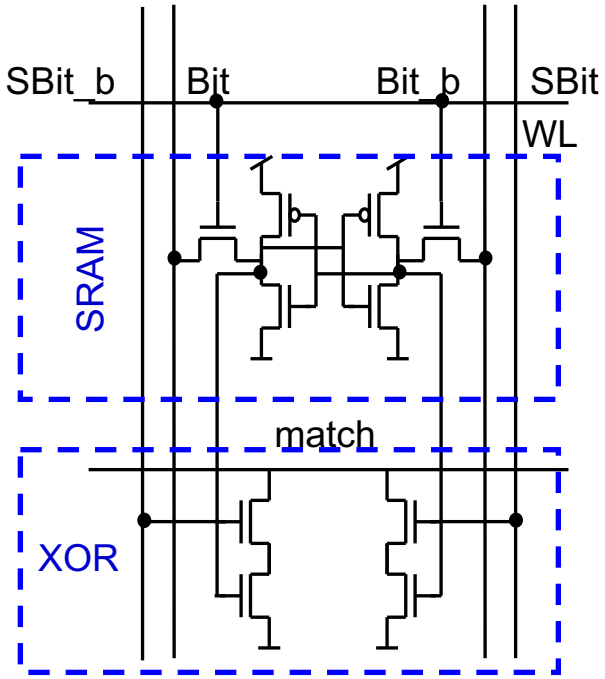
Content Addressable Memory

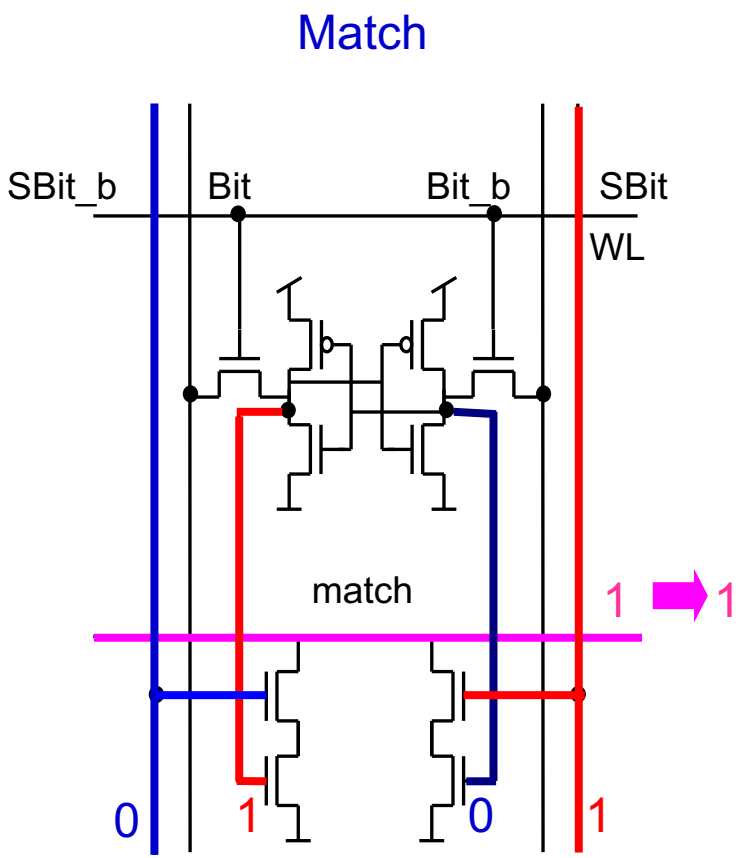
- Commonly used in translation lookaside buffers (TLBs).
- Matching asserts a matchline output for each world that contains a specified key

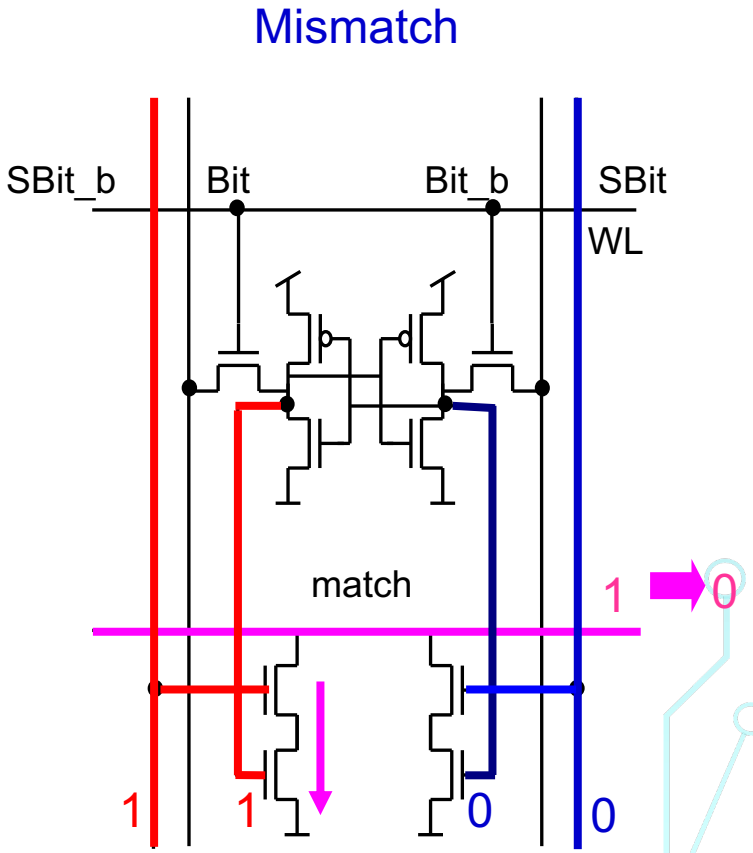


Content Addressable Memory

- Commonly used in translation lookaside buffers (TLBs).
- Matching asserts a matchline output for each world that contains a specified key



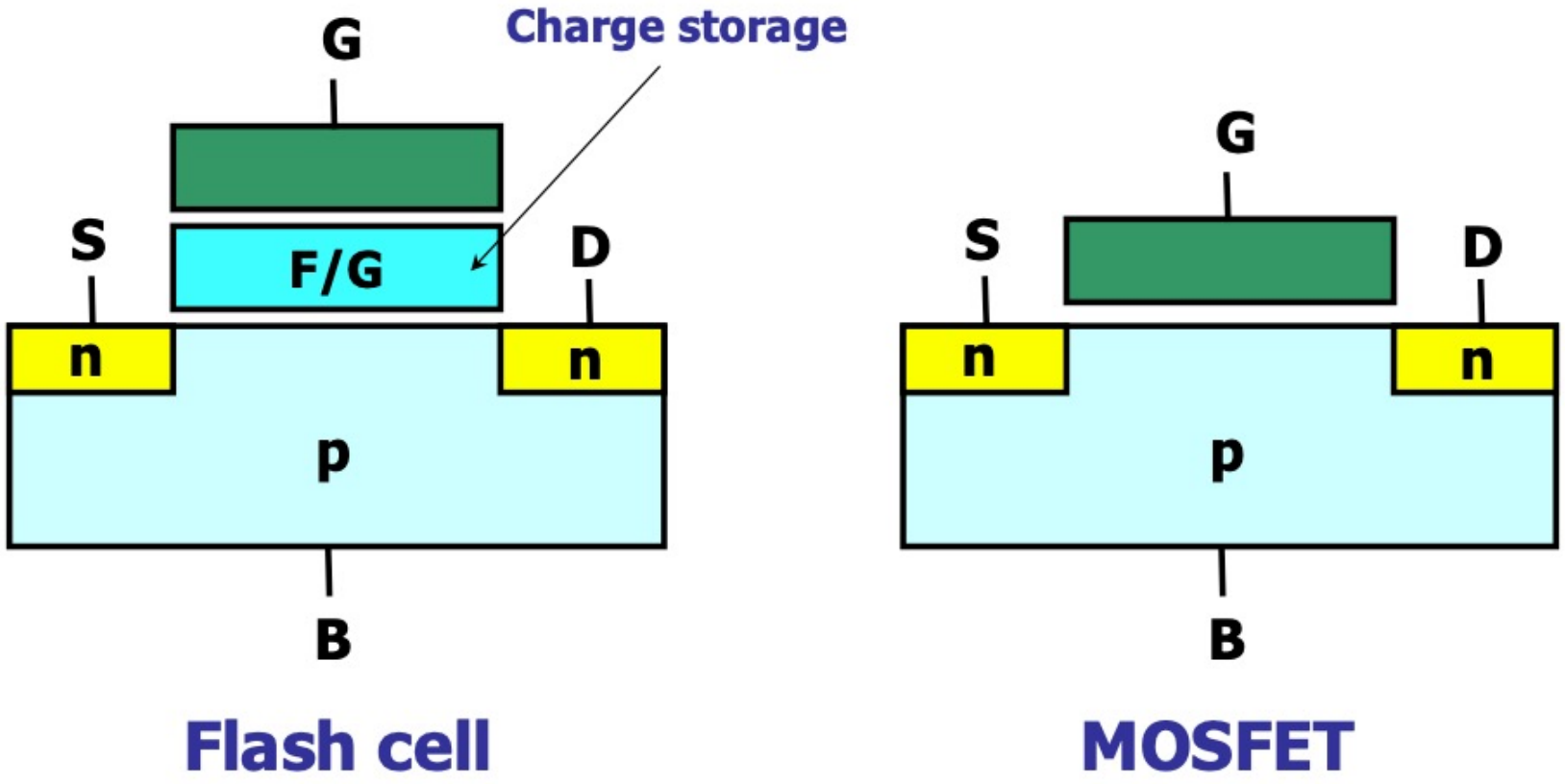




- Cache
- DRAM
- Content-Addressable Memory
- Flash

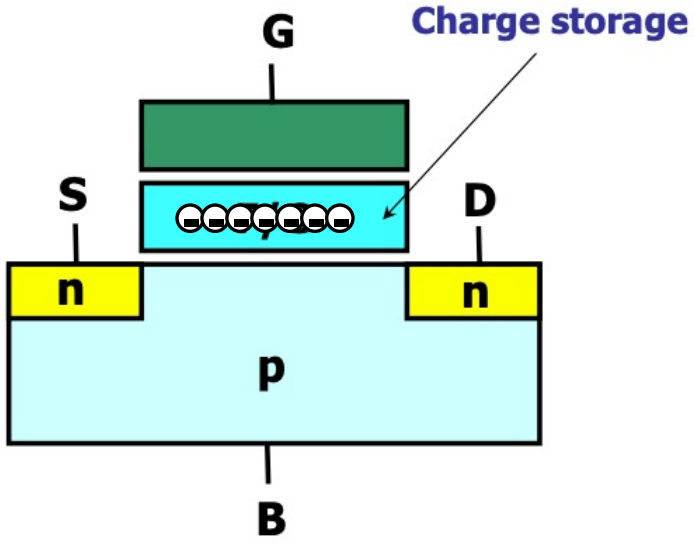
Key concept: Floating Gate

- Floating Gate: A charge storage layer -> memorize information
- A “Programmable-Threshold” Transistor

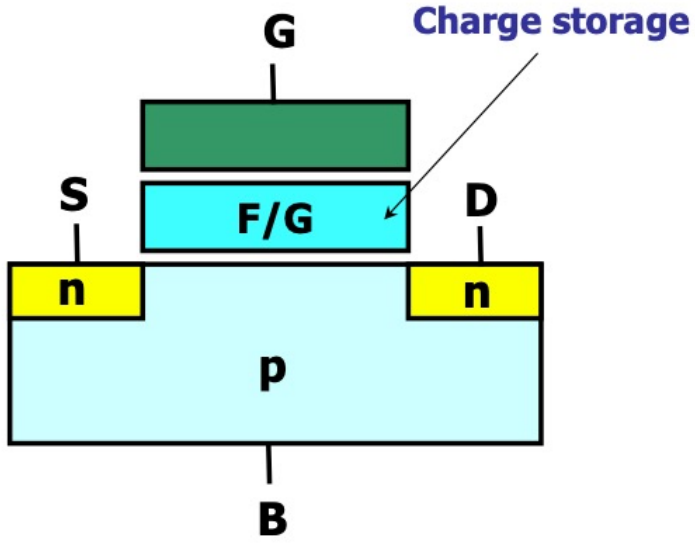


Read a Flash cell

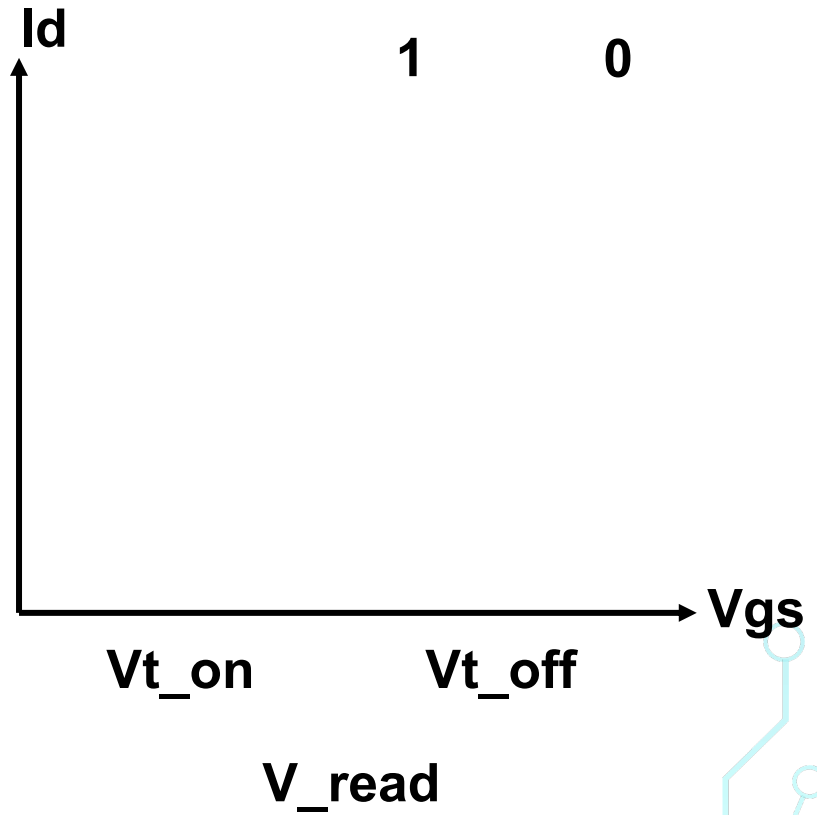
- Floating gate change the threshold voltage of a cell
- Read the cell value by sensing the current



Storing 0
OFF State

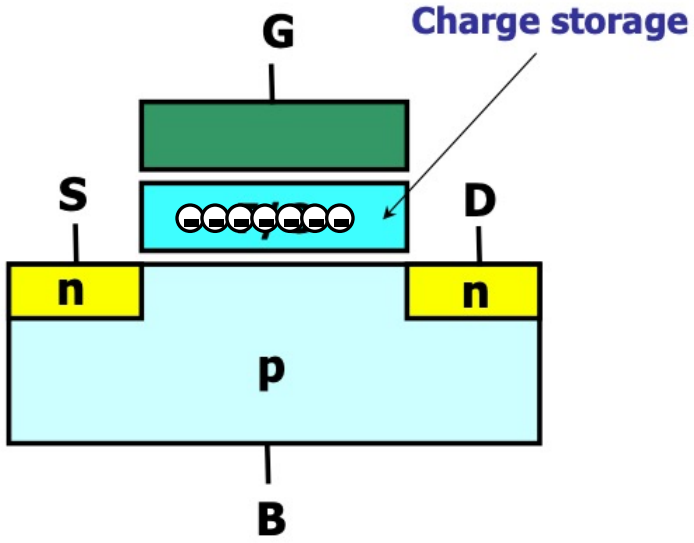


Storing 1
ON State

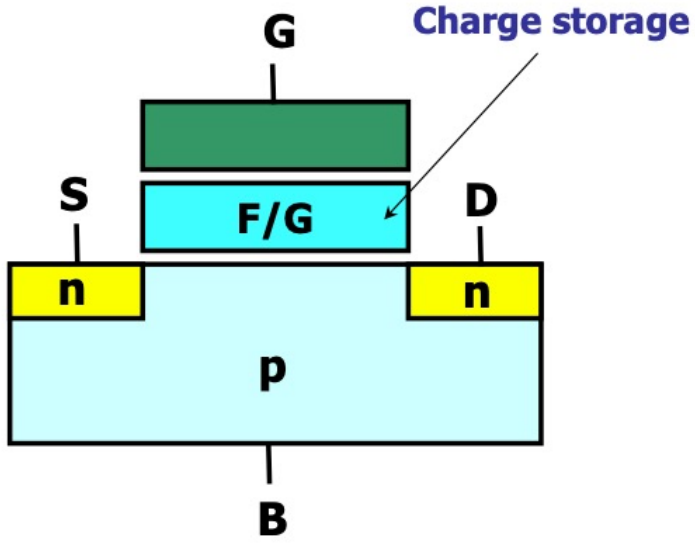


Read a Flash cell

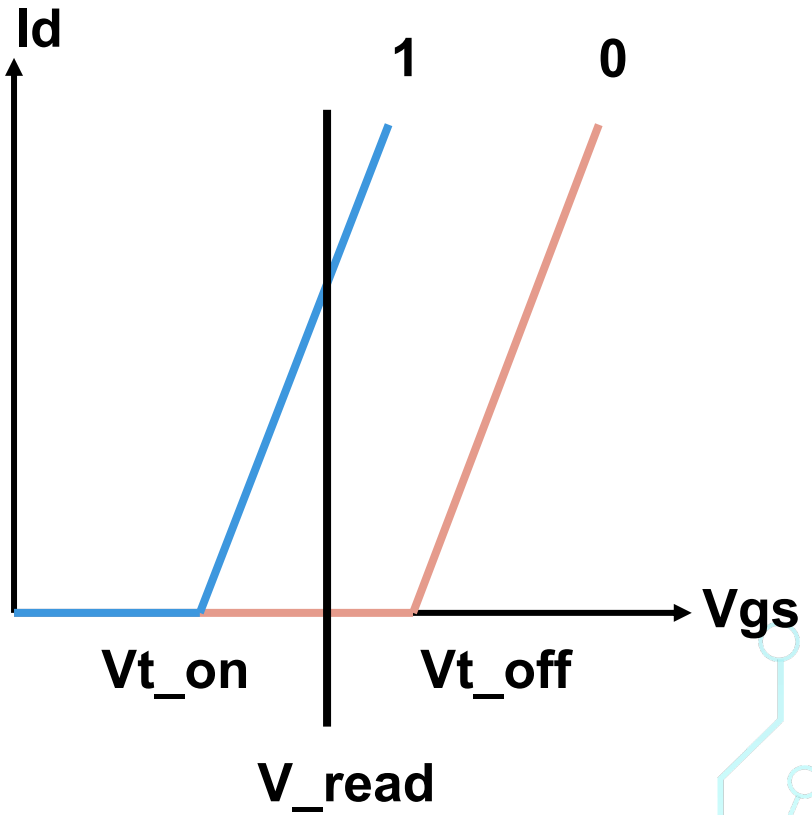
- Floating gate change the threshold voltage of a cell
- Read the cell value by sensing the current



Storing 0
OFF State

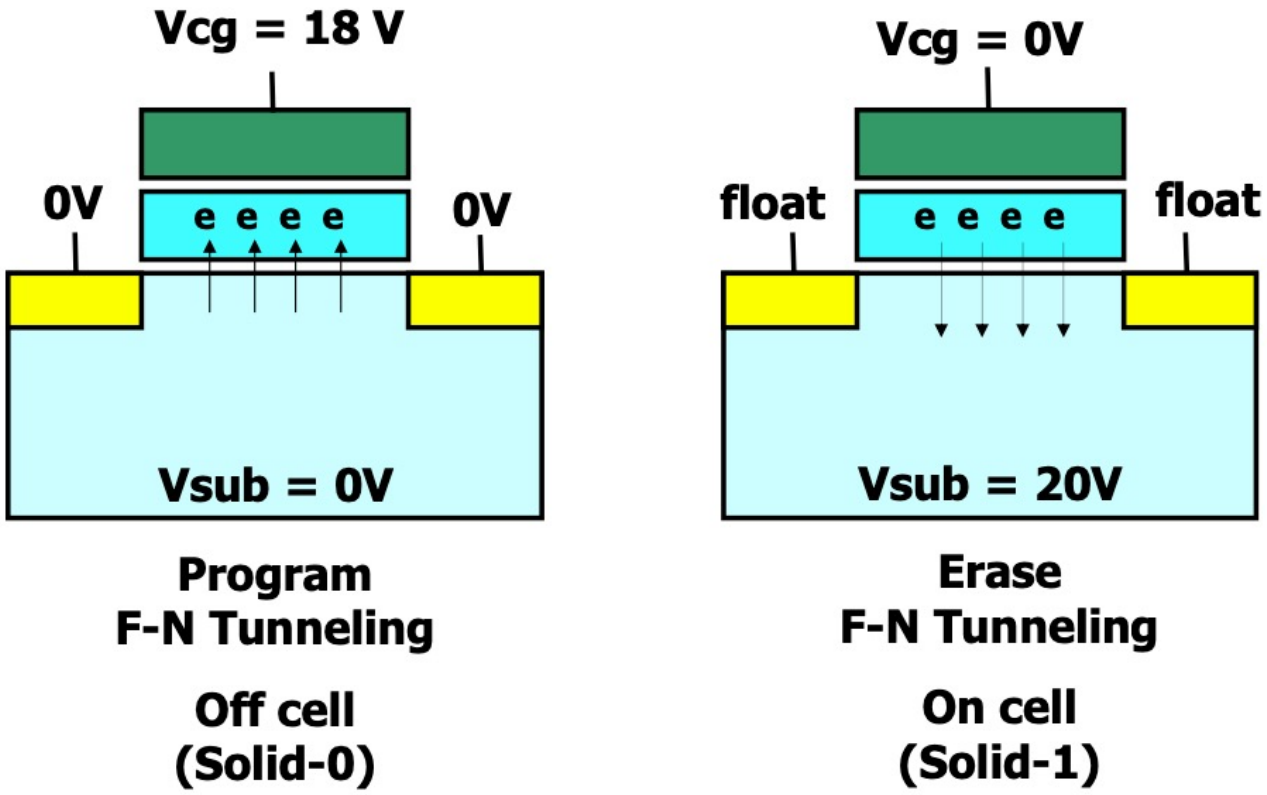


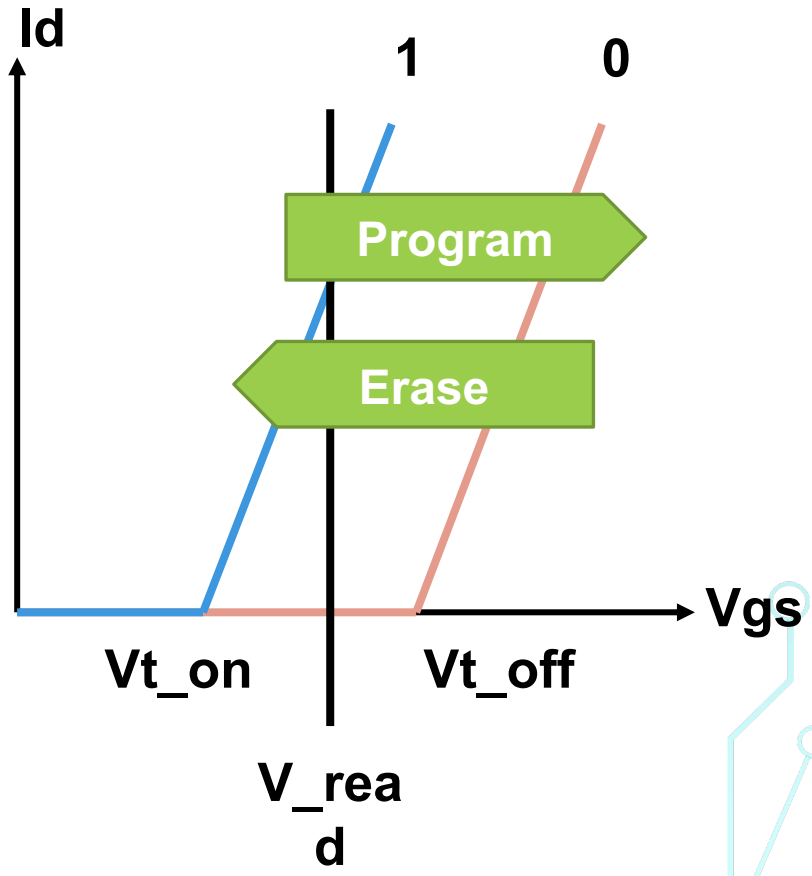
Storing 1
ON State



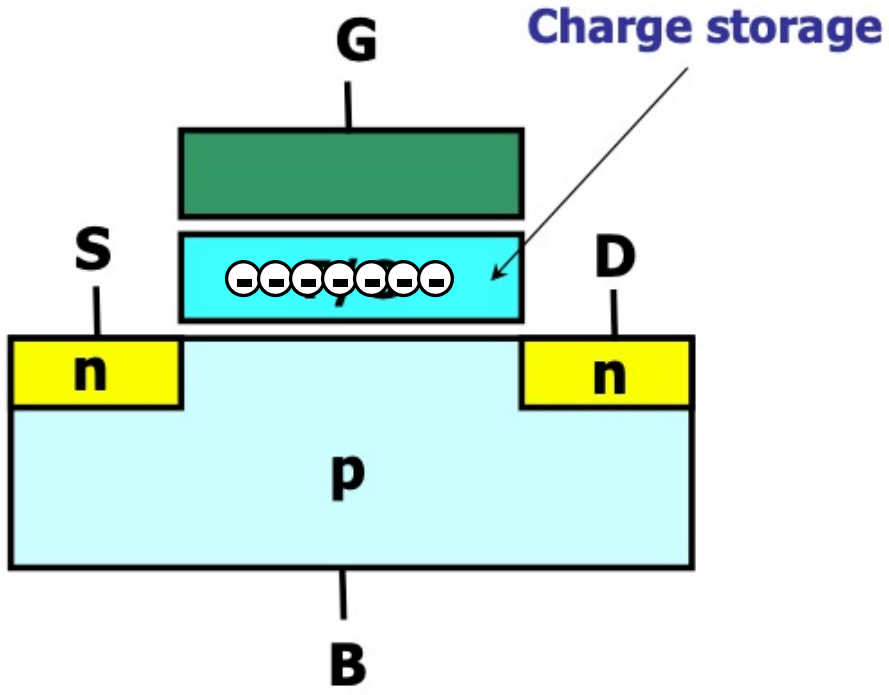
Write a Flash cell

- Write 0: program; Write 1: erase
- Must be erased (storing 1) before reprogrammed.
- Endurance: ~100K erase-program cycles

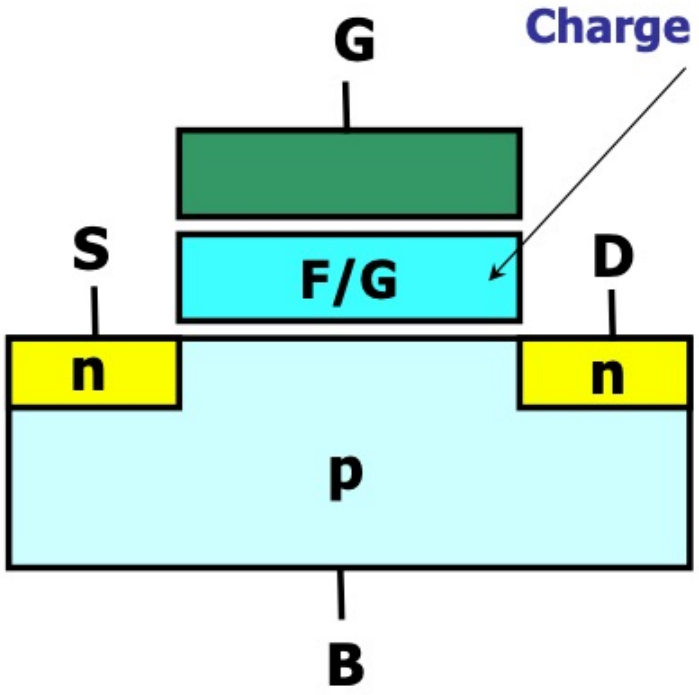




Single-Level Cell: 0 and 1 in Flash



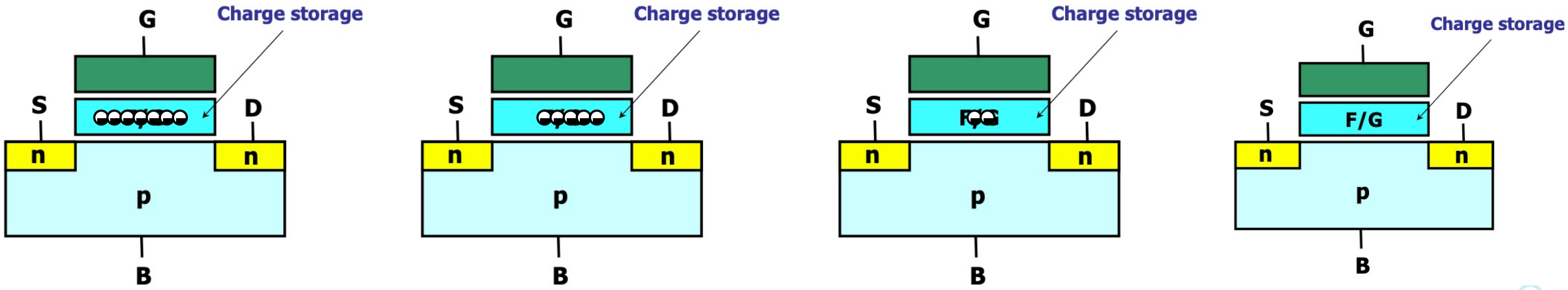
- Storing 0
- Negative charge in floating gate



- Storing 1
- No charge in floating gate

Multi-Level Cell

- Higher density
- More errors



Review

- Memory arrays:
  - SRAM:
    - Unique combination of density, speed, power
    - SRAM cell: stability and writeability
  - Caches
    - Direct mapped and set-associative
  - DRAM
    - 1-T volatile
  - Content-Addressable Memory (CAM)
    - SRAM cell + XOR
  - Flash
    - Floating gate