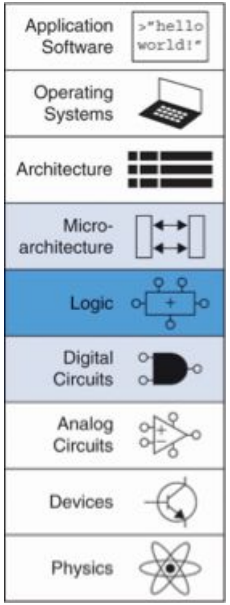
COSC175 (Systems I): Computer Organization & Design

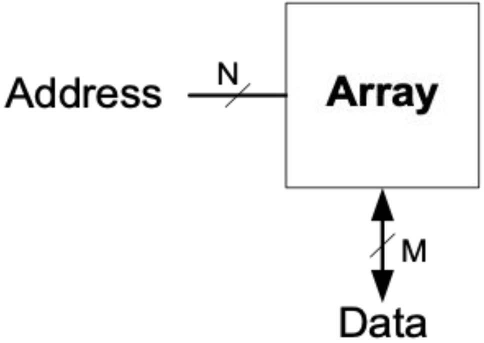
Professor Lillian Pentecost
Fall 2024

Warm-Up October 8

- Where we were
  - ALUs, practice with number representations, etc.
- Where we are going
  - Memory Arrays; filling in our last important building block!
- Logistics, Reminders
  - TA help 7-9PM on Sundays, Tuesdays, Thursdays in C107
  - LP Office hours M 9-10:30AM, Th 2:30-4PM
  - Weekly Exercises Due Friday 5PM (CYOA)
  - ***Pre-Lab for tomorrow*** is to bring your Lab 4: First Stage: Part 2 diagram of your ALU!
    - We will walk through example tests in detail (including some anon. requests)
  - ***Midterm Exam Announcement: Tuesday October 22, 6PM (see Day9 for details)***
- Textbook Tags: 5.5



Memory Arrays

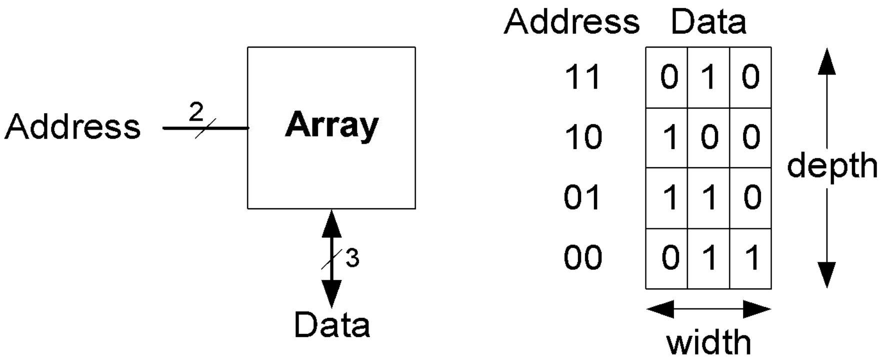


- Efficiently store large amounts of data
- *M*-bit data value read/written at each unique *N*-bit address
- Remembering our processor architecture, filling in micro-architectural details

"Memory is the universal elixir of the computing system."

- Fred Brooks, 2018

Memory Array Example

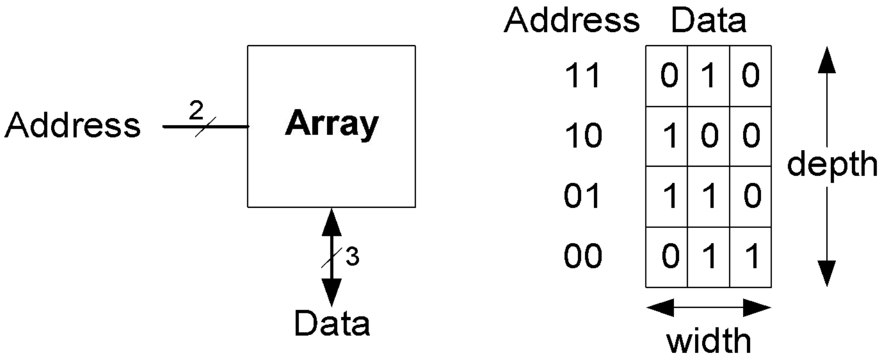


Address	Data		
11	0	1	0
10	1	0	0
01	1	1	0
00	0	1	1

- **2^2 × 3-bit** array
- **Number of words:** 4
- **Word size:** 3-bits
- For example, the 3-bit word stored at address 10 is 100

Memory Array Example 2

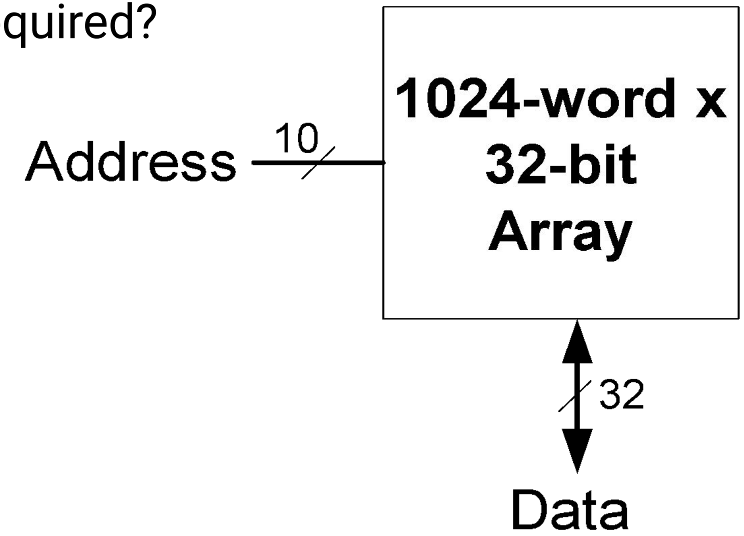
- For a memory array with ***1024 words***, how long of an address (i.e., how many bits) are required?



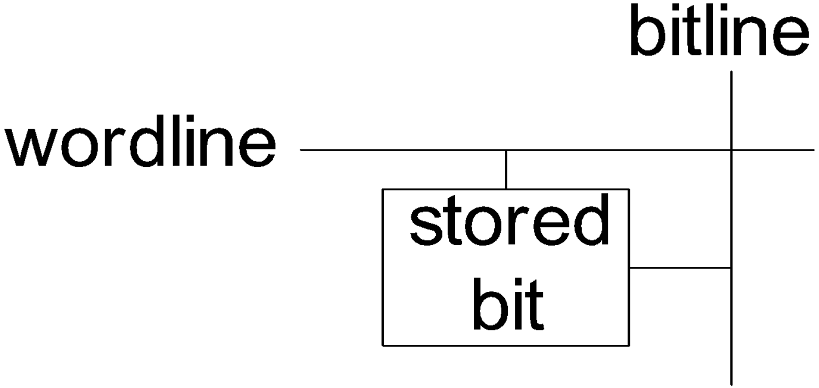
Address	Data		
11	0	1	0
10	1	0	0
01	1	1	0
00	0	1	1

Memory Array Example 2

- For a memory array with ***1024 words***, how long of an address (i.e., how many bits) are required?

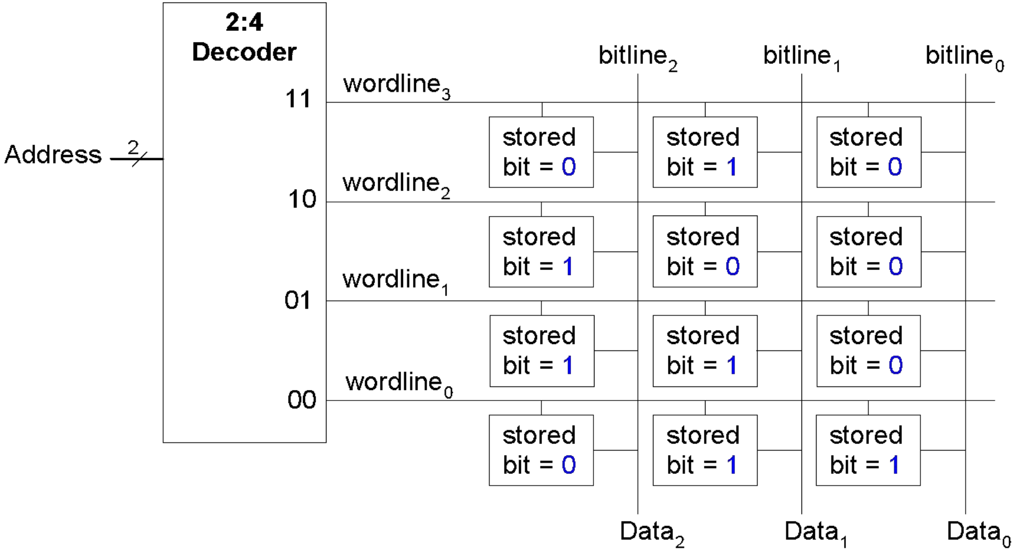


Memory Arrays: Bit Cells



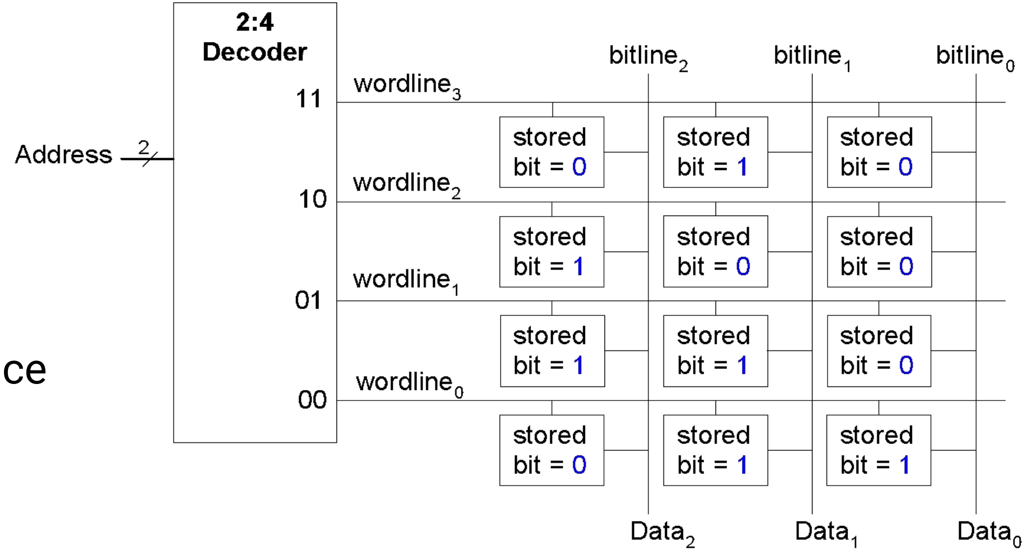
Memory Array: Wordlines & Bitlines

- An example array:
  - 4 words, 3 bits per word
- A ***decoder*** from input address to set one wordline as 1's
- Horizontal wordlines
- Vertical Bitlines



Wordlines & Bitlines

- **Wordline:**
  - like an enable
  - single row in memory array read/written
  - corresponds to unique address
  - only one wordline 1/**HIGH** at once



2:4
Decoder

Address
2

11
10
01
00

bitline$_2$
bitline$_1$
bitline$_0$

wordline$_3$

stored bit = 0
stored bit = 1
stored bit = 0

wordline$_2$

stored bit = 1
stored bit = 0
stored bit = 0

wordline$_1$

stored bit = 1
stored bit = 1
stored bit = 0

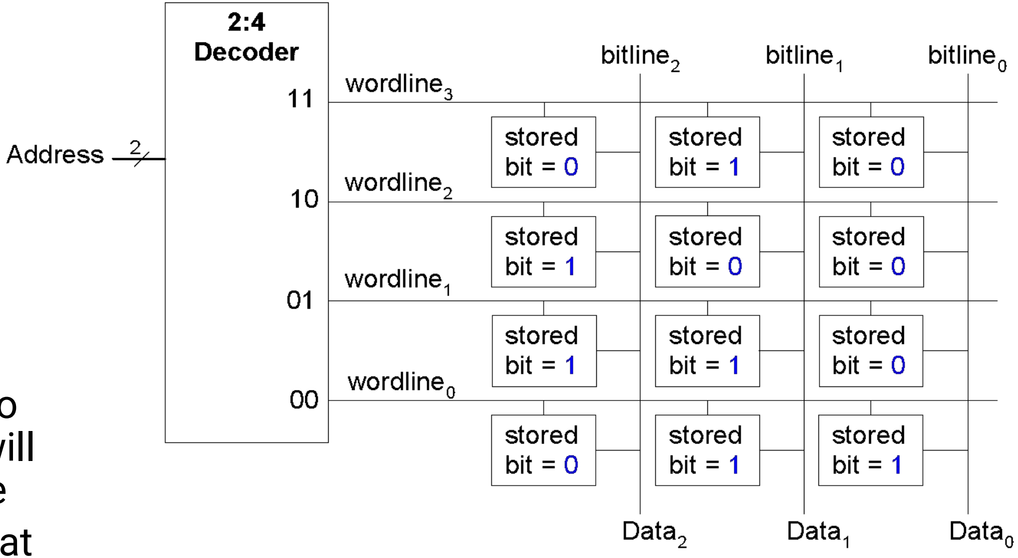
wordline$_0$

stored bit = 0
stored bit = 1
stored bit = 1

Data$_2$
Data$_1$
Data$_0$

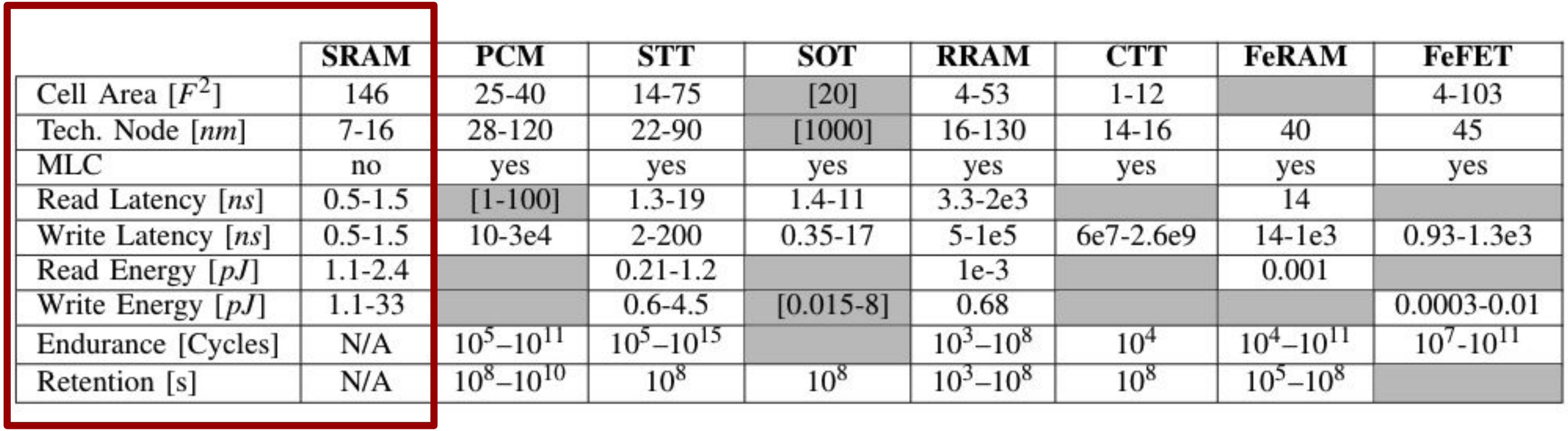
Wordlines & Bitlines

- **Bitline:**
  - **floating** when not enabled
  - when enabled, stored bit transfers to/from the bitline
  - for a **read**, Data takes value of enabled row across all bitlines
  - for a **write**, each bitline will be activated with the desired values to store, then activating a worldline will force the stored bits into that state
  - the physical “how” depends on what ***memory technology*** and type is used



Types of Memory

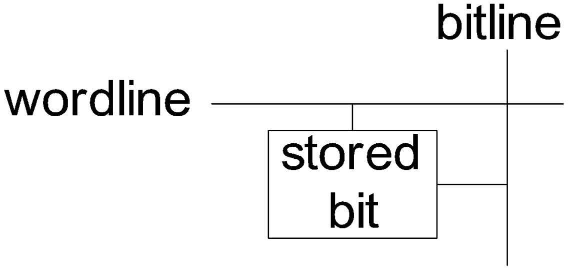
- Memory arrays encode or store bits using different physical means
- Sometimes ***volatile*** (data is lost when powered off), sometimes ***non-volatile***
- **RAM** stands for “Random Access Memory”
    - Traditional definition: the latency (time) to access any memory location is roughly the same

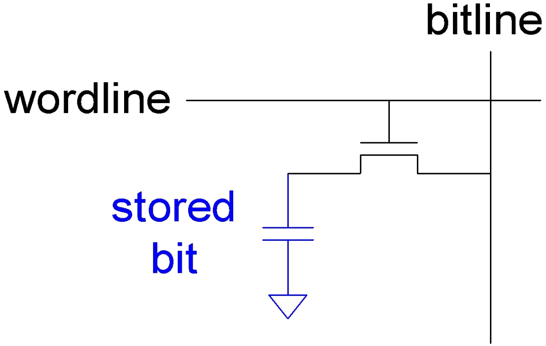


	SRAM	PCM	STT	SOT	RRAM	CTT	FeRAM	FeFET
Cell Area [F^2]	146	25-40	14-75	[20]	4-53	1-12		4-103
Tech. Node [nm]	7-16	28-120	22-90	[1000]	16-130	14-16	40	45
MLC	no	yes	yes	yes	yes	yes	yes	yes
Read Latency [ns]	0.5-1.5	[1-100]	1.3-19	1.4-11	3.3-2e3		14	
Write Latency [ns]	0.5-1.5	10-3e4	2-200	0.35-17	5-1e5	6e7-2.6e9	14-1e3	0.93-1.3e3
Read Energy [pJ]	1.1-2.4		0.21-1.2		1e-3		0.001	
Write Energy [pJ]	1.1-33		0.6-4.5	[0.015-8]	0.68			0.0003-0.01
Endurance [Cycles]	N/A	10^5–10^{11}	10^5–10^{15}		10^3–10^8	10^4	10^4–10^{11}	10^7-10^{11}
Retention [s]	N/A	10^8–10^{10}	10^8	10^8	10^3–10^8	10^8	10^5–10^8	

Robert Dennard, 1932 - 2024

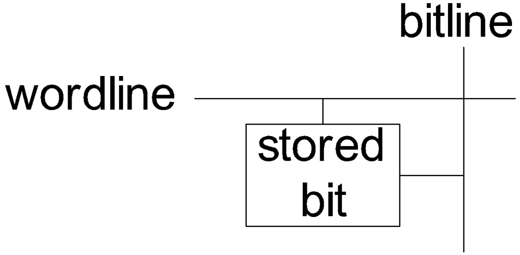
- Invented **DRAM** in 1966 at IBM
- Others were skeptical that the idea would work
- By the mid-1970's DR
- Each bit in DRAM is stored using a *capacitor*
- *D is for **Dynamic*** because the value needs to be **refreshed** (rewritten) periodically and after read:
  - Charge leakage from the capacitor degrades the value
  - Reading destroys the stored value

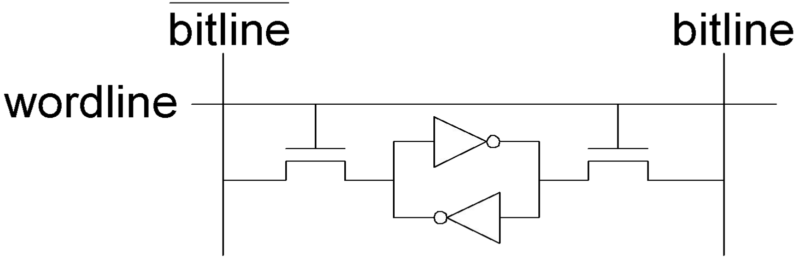




SRAM is faster than DRAM

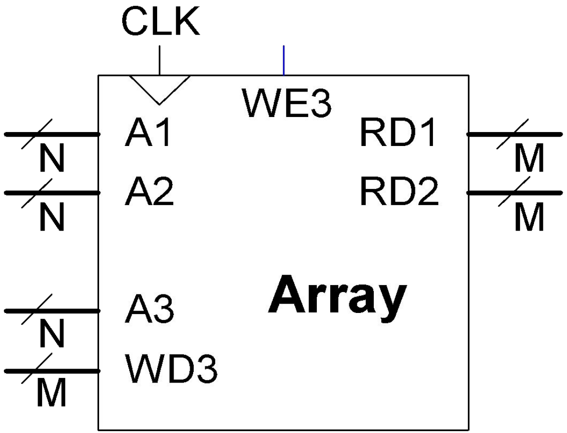
- Another ***volatile*** memory
- *S for "Static"*, no refresh needed
- Bitcell design is larger, more complex per-bit than DRAM





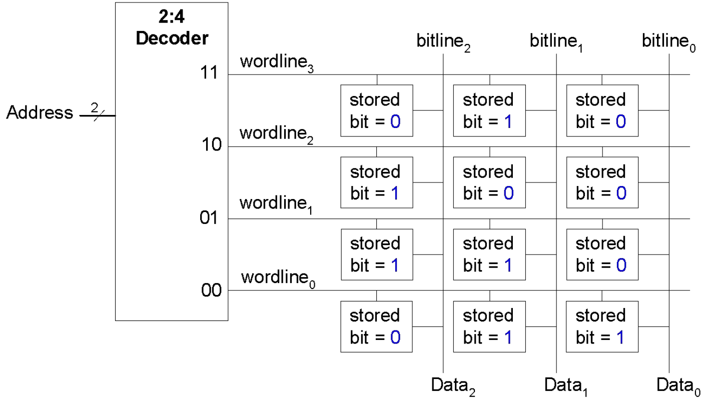
Multi-ported Memories

- **Port:** address/data pair
- 3-ported memory
  - 2 read ports (A1/RD1, A2/RD2)
  - 1 write port (A3/WD3, WE3 enables writing)
- **Register file:** small multi-ported memory



Memory Arrays Review

Memory Type	Transistors per Bit Cell (Area)	Latency (Time)
Flip-Flop	~20	Fastest!!
SRAM	6	Still Fast
DRAM	1 (+1C)	Slower



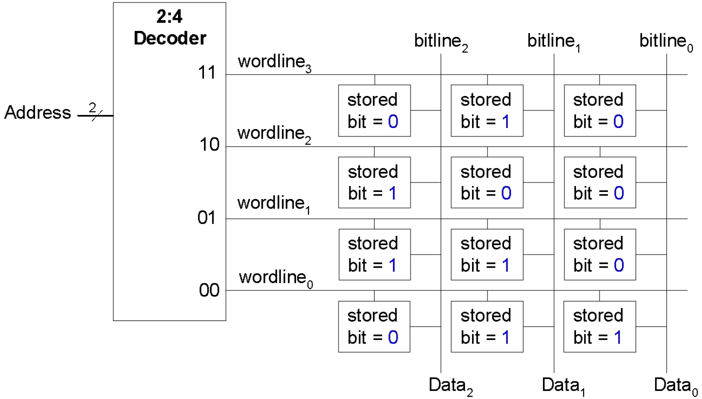
- SRAM is faster, and uses only the same building blocks as logic, so is used for memory arrays closest to processing (like the registerfile)
- DRAM achieves higher capacity, and hangs out nearby for the rest of our program data
- But these are all ***volatile*** technologies, what happens when we turn our computer off?

Fujio Masuoka, 1944 -

- Developed memories and high speed circuits at Toshiba, 1971-1994
- Invented ***Flash*** memory as an unauthorized project pursued during nights and weekends in the late 1970’s
- The process of erasing the memory reminded him of the flash of a camera
- Toshiba slow to commercialize the idea; Intel was first to market in 1988
- Flash has grown into a $25 billion per year market

Memory Arrays Review

Memory Type	Transistors per Bit Cell (Area)	Latency (Time)
Flip-Flop	~20	Fastest!!
SRAM	6	Still Fast
DRAM	1 (+1C)	Slower

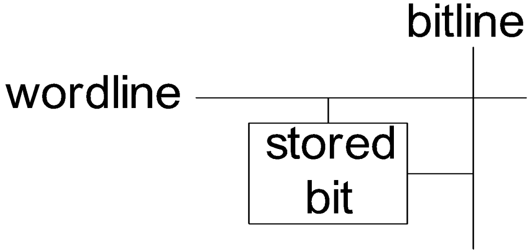


- We will eventually build out a system that utilizes SRAM, DRAM, and non-volatile memory technologies in order to support efficient data storage & data movement!

“Memory is the universal elixir of the computing system.”

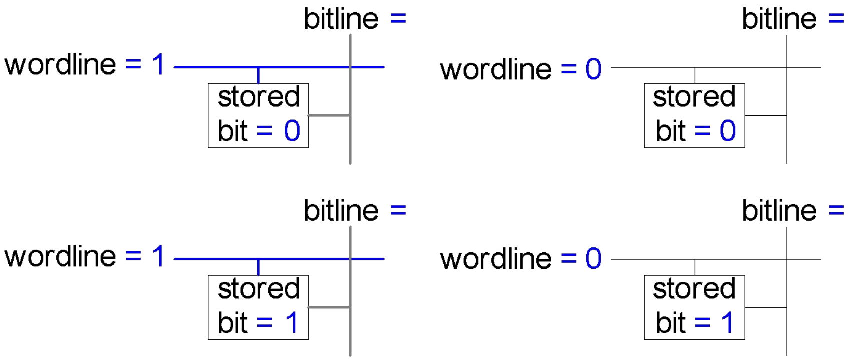
- Fred Brooks, 2018

Check-In Activity: Bit Cells



Individually or with a neighbor, on a notecard:

Part A: for each of the four bitcell settings, what value would the bitline read out to the data?



Part B: What questions do you have about types of memory, memory arrays, or the history of computing? Discuss with a partner and write down any details that were unclear today, or questions you may have compared to what you have read or heard about memory types in laptops, smartphones, or other devices.

Wrap-Up October 8

SCAN ME

- Coming up next!
  - Putting our pieces together, and the history of computing hardware!
- Logistics, Reminders
  - TA help 7-9PM on Sundays, Tuesdays, Thursdays in C107
  - LP Office hours M 9-10:30AM, Th 2:30-4PM
  - Weekly Exercises Due Friday 5PM (CYOA)
  - Lab 4 Report due **October 17 10PM**
  - ***Pre-Lab for tomorrow*** is to bring your Lab 4: First Stage: Part 2 diagram of your ALU!
    - We will walk through example tests in detail (including some anon. requests)
  - ***Midterm Exam Announcement: Tuesday October 22, 6PM (see Day9 for details)***
- FEEDBACK
  - https://forms.gle/5Aafcm3iJthX78jx6