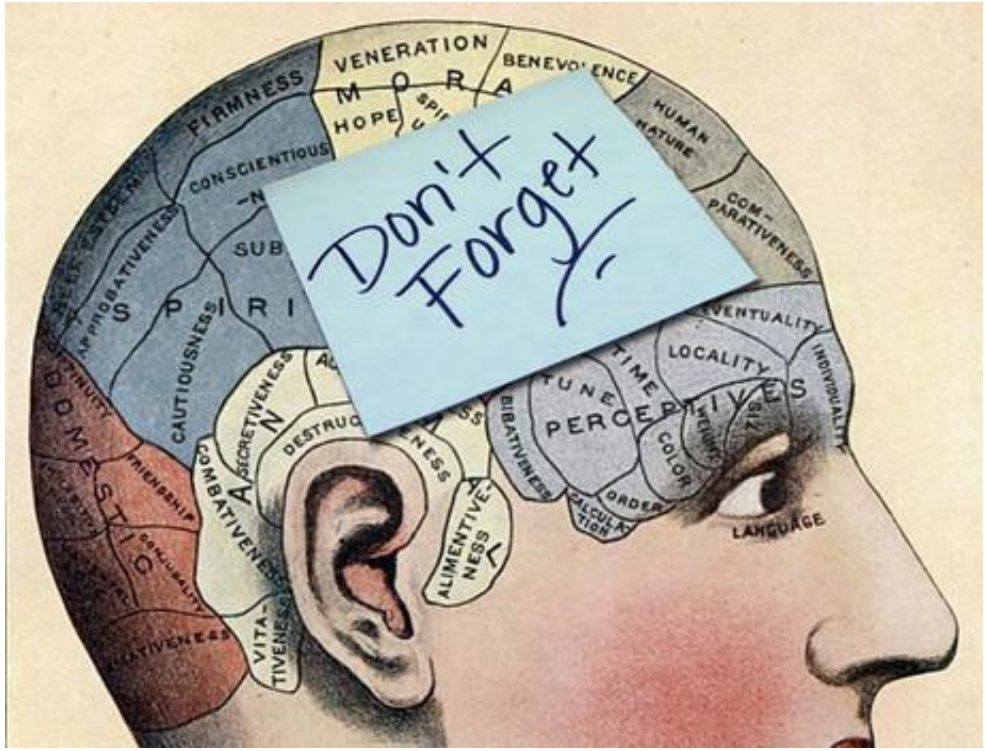
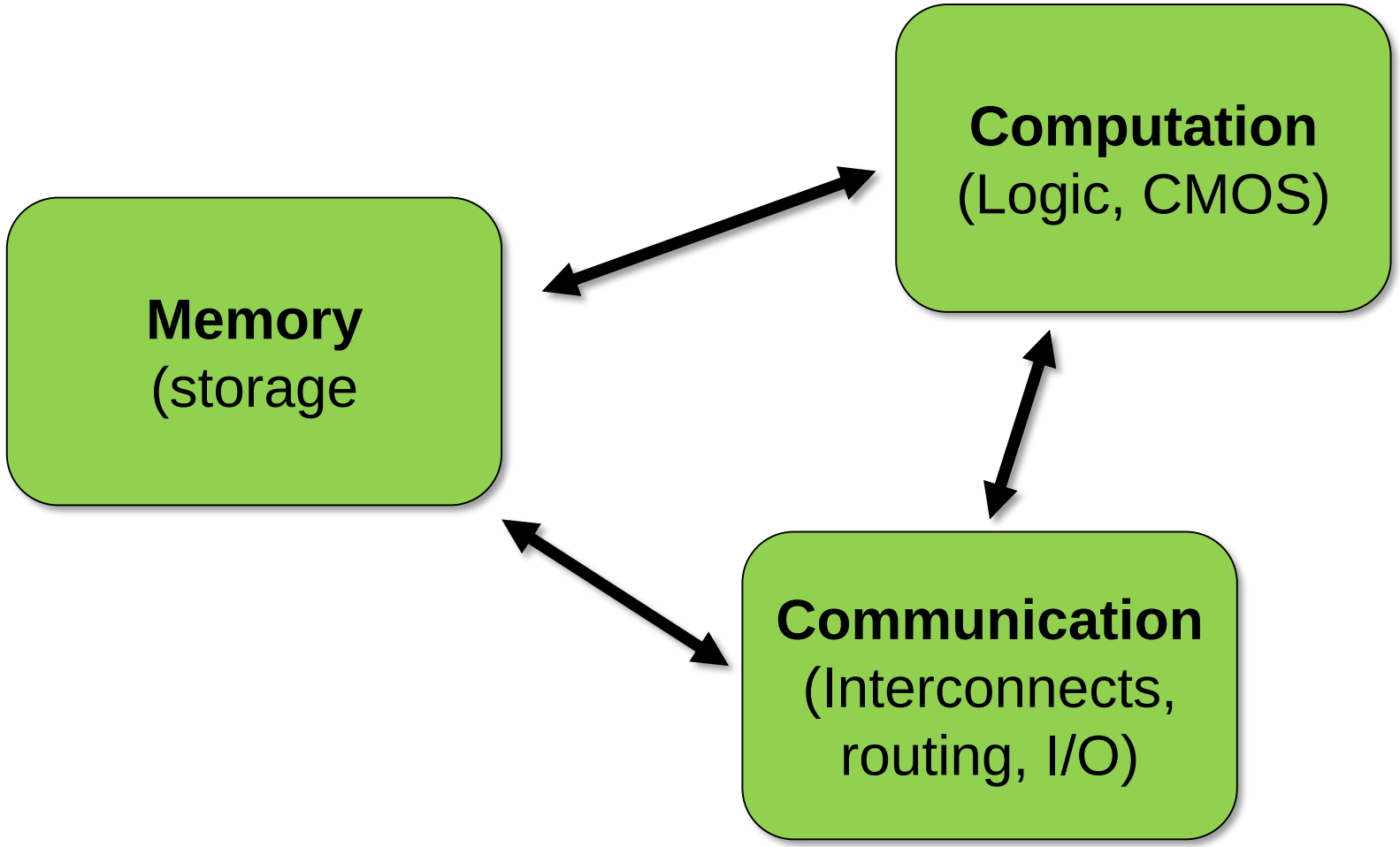
Lecture 1 - Introduction

Outline – Lecture 1

- Why memory?
- Opportunities for emerging memory for storage
- Opportunities for memory devices in machine learning

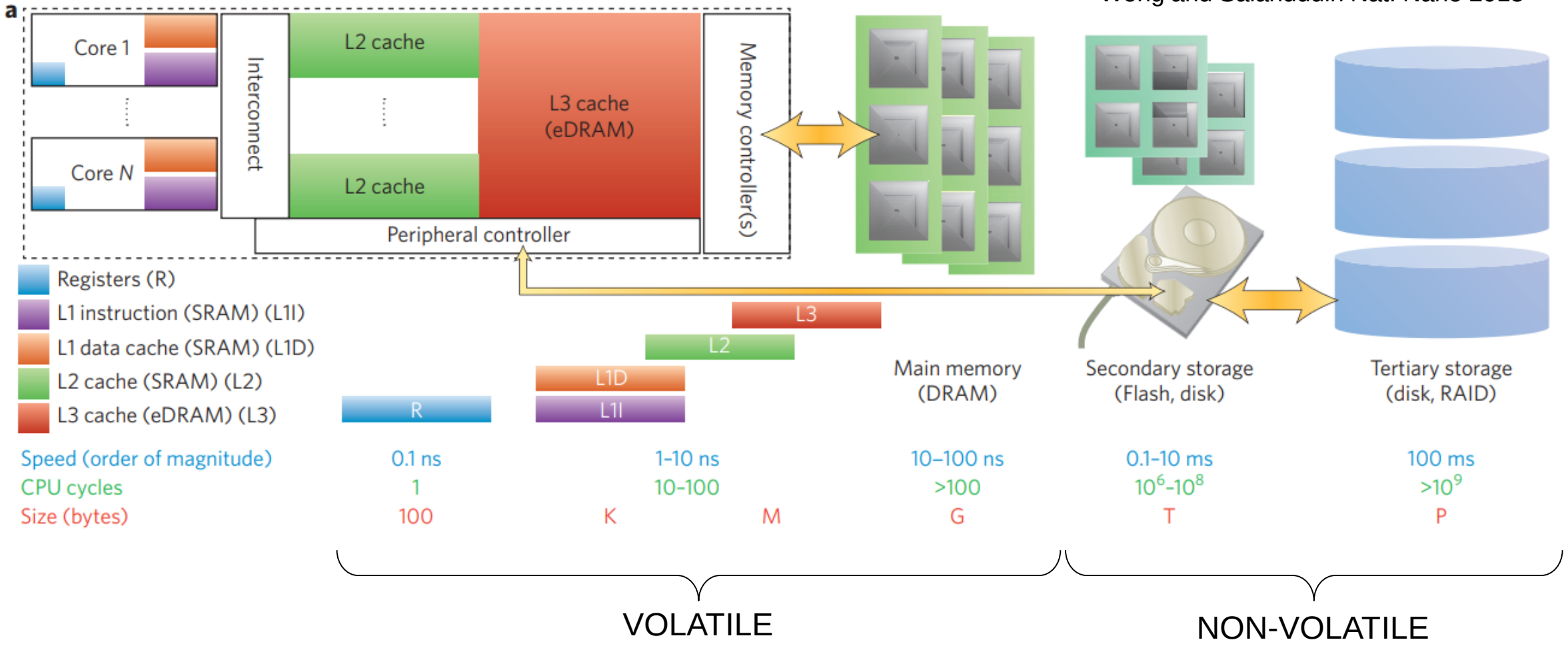


Why bother about Memory?



Memory hierarcy

Wong and Salahuddin Nat. Nano 2015



Modern work-loads

- Modern work-loads are data-driven

Modern work-loads require more memory

- Big data → big memory
- Big data is shuffled around → Need fast data access

- Harddrive bandwidth has not kept up with capacity!

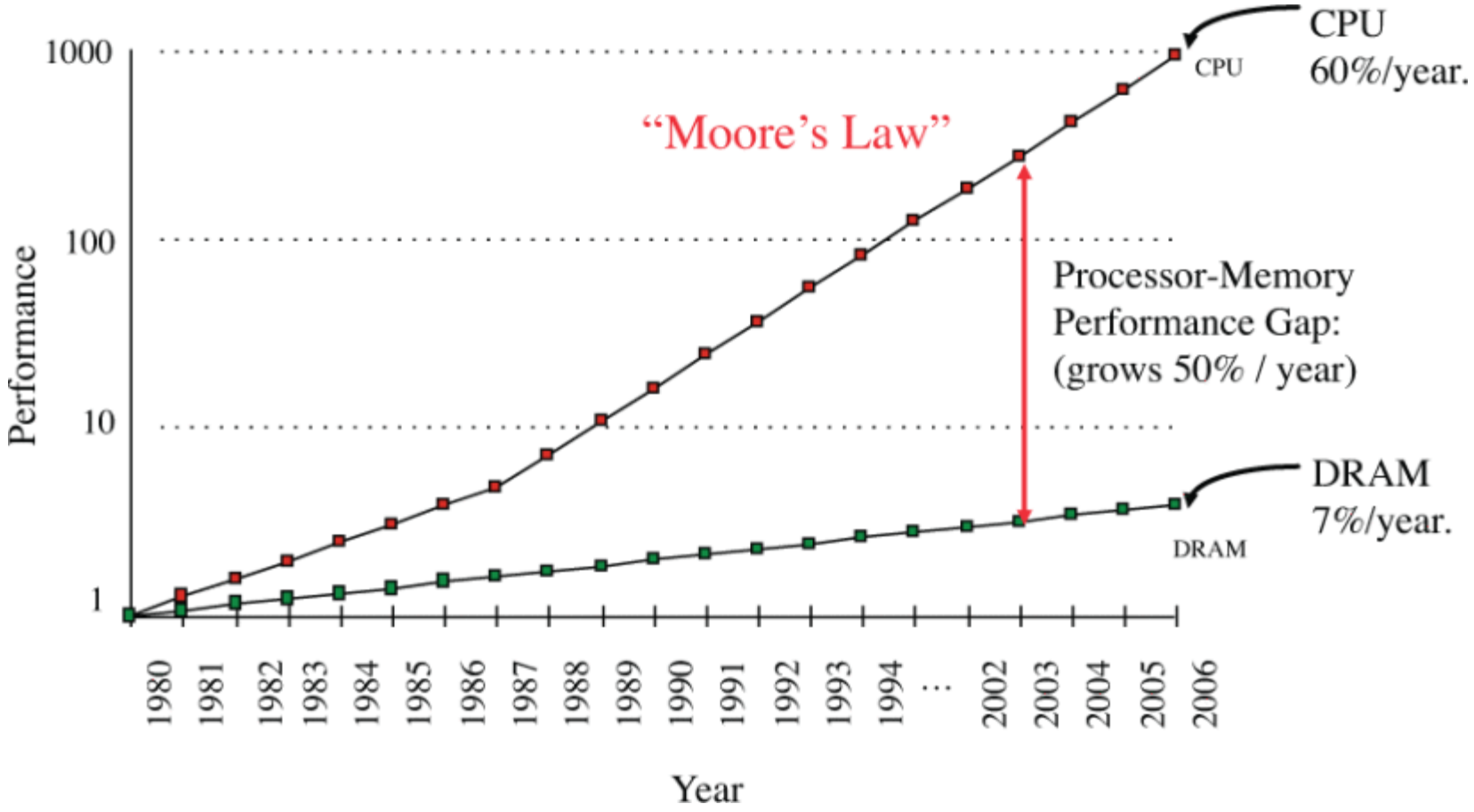
	1985	2019	Change
Capacity	30 MB	15 TB	**500 000x**
Max. Transfer Rate	2 MB/s	260 MB/s	130x
Latency	20 ms	4.17 ms	5x
Capacity/MTR	15 s	**57700 s (16h!)**	**3846x**

- SSDs better, but not much: MTR ~ 2.5 GB/s → Cap/MTR ~ 200s

- **Q: What can be a solution to this problem?**

The memory gap

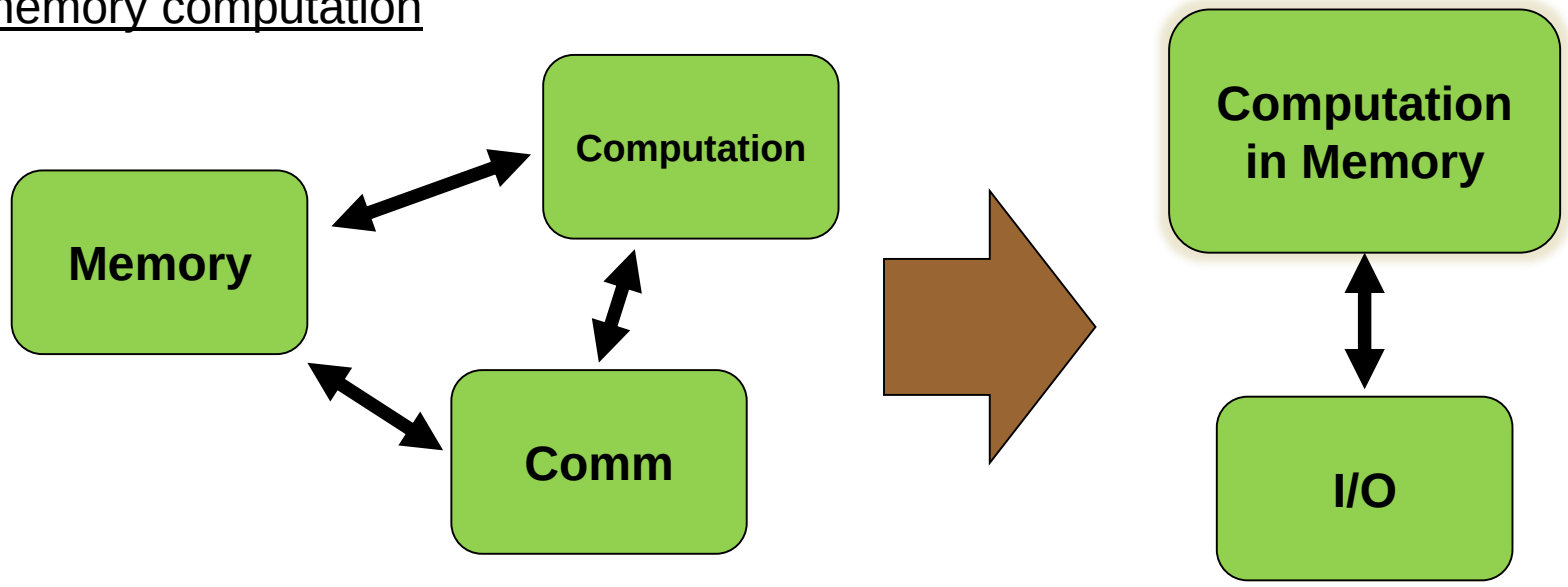
- Von-Neumann bottleneck, even on-chip doesn't always help!



10.1109/SBAC-PAD.2011.10 Bahi & Eisenbeis 2011

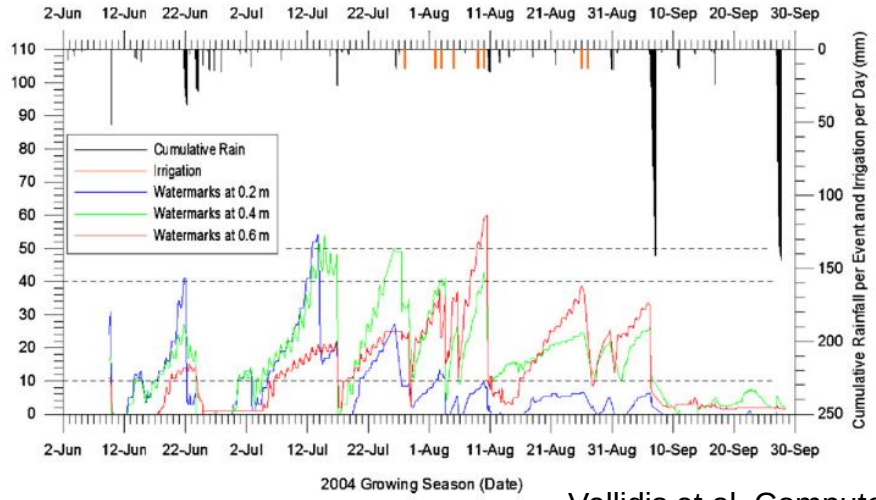
In-memory computation

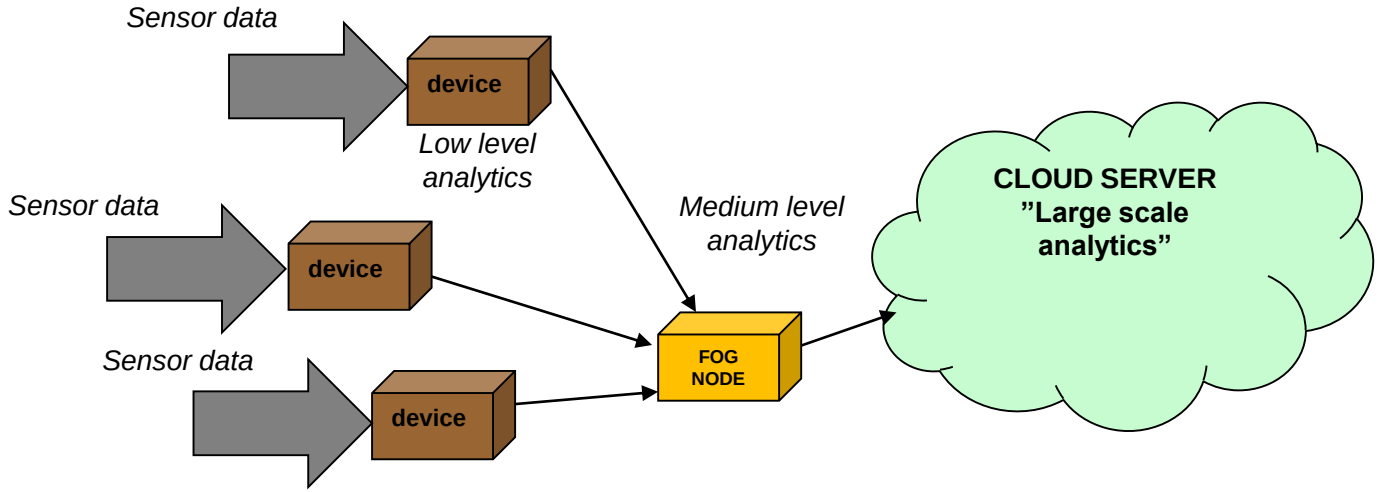
- Instead of moving memory to the logic → move the computation into the memory
- Moving memory takes resources
- Use properties of non-volatile memory to perform calculations directly in the memory circuit!
- Called In-memory computation



Computing at the edge, power constraints

Sensor node

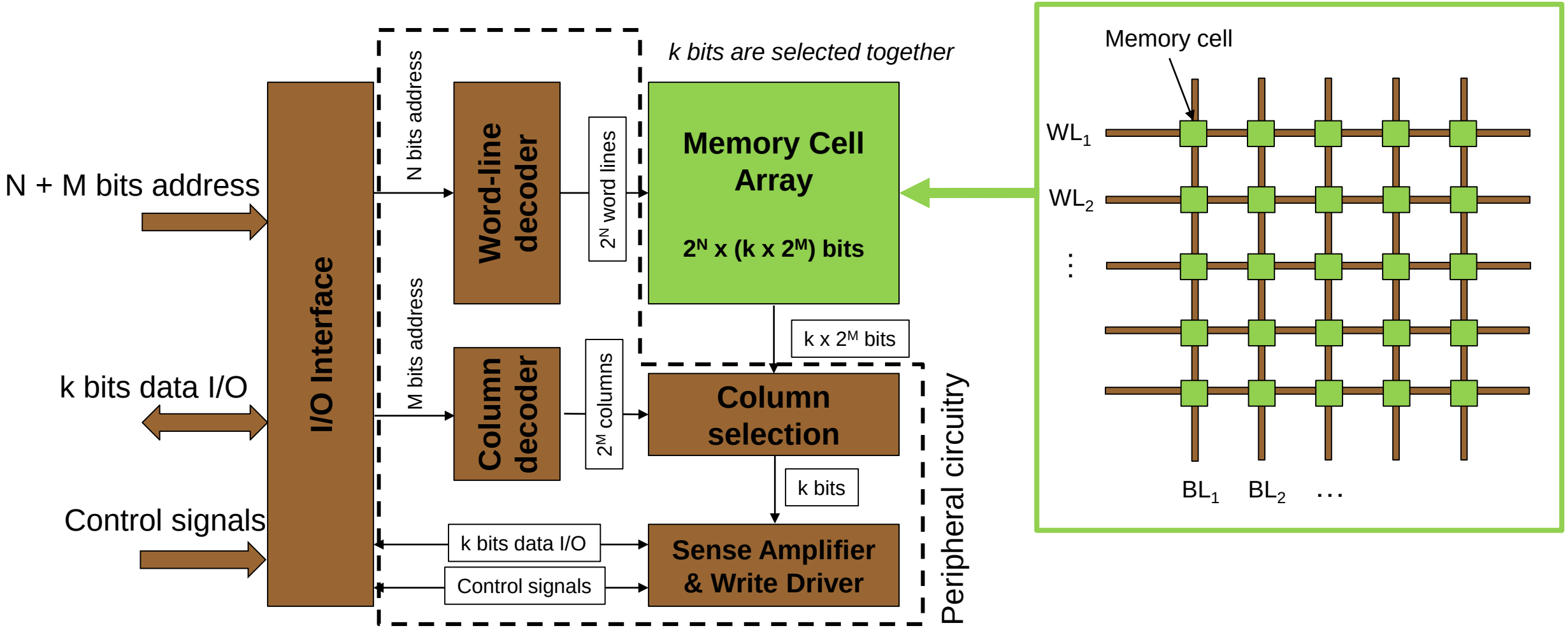




- More functionality is pushed to the edge
  - Battery powered devices
- Memory cannot use up as much energy as today

Vellidis et al. Computers & Electronics in Agriculture 2008

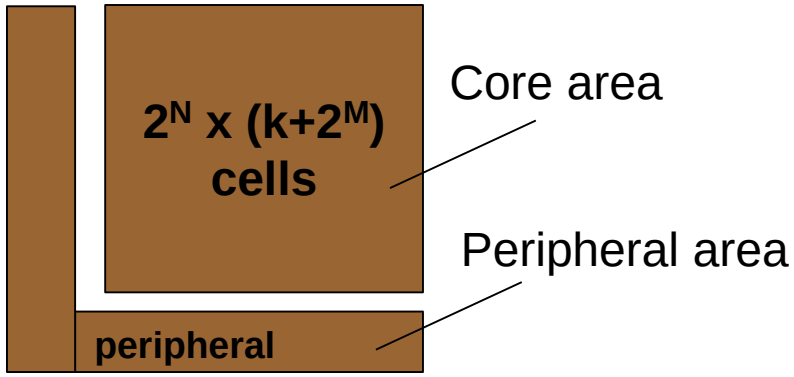
Generic architecture of memory arrays



Array Area efficiency

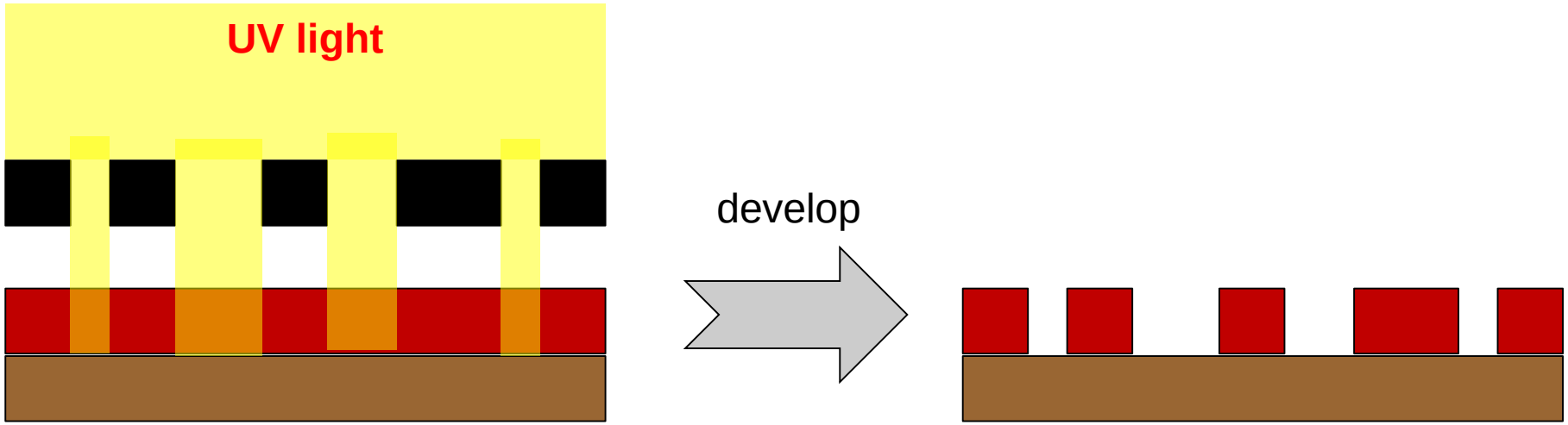
- Core Area (Memory Array area)
  - How many memory cells can fit in a certain area?
- Peripheral Area
  - Sense amplifier, decoders, multiplexer, etc..
  - How much area is needed for peripheral circuits?

- Area Efficiency = Core / (Core + Peripheral Circuits) < 1



How small can a memory element be?

- Device size is limited by the patterning technology → Technology node
- Smallest ”feature size” defined as F.

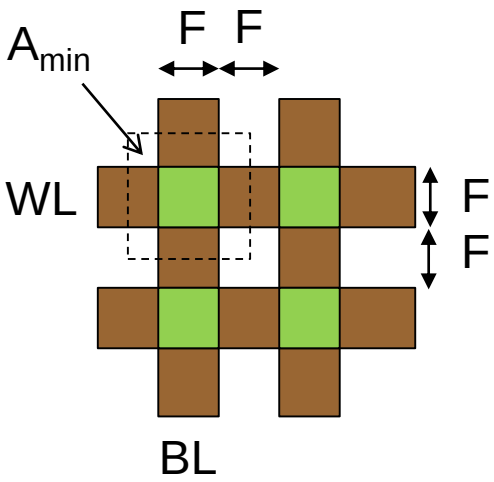


Q: What limits resolution?

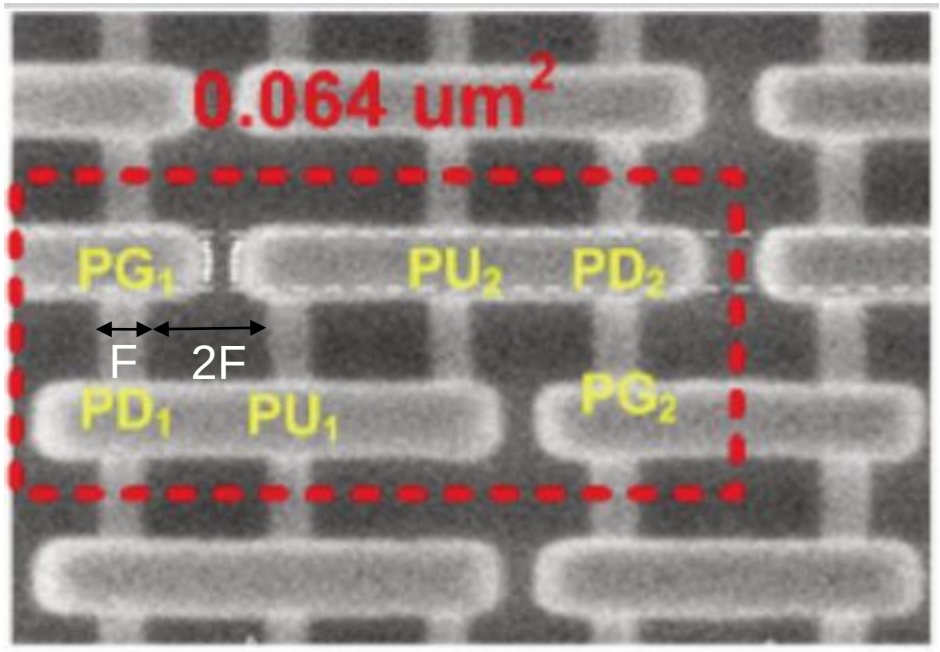
Area scaling

- Feature size F depends on technology node
  - Smallest possible feature that can be fabricated
- Ideal 2D memory cell size: $A_{min} = 4F^2$. **Q: Why?**

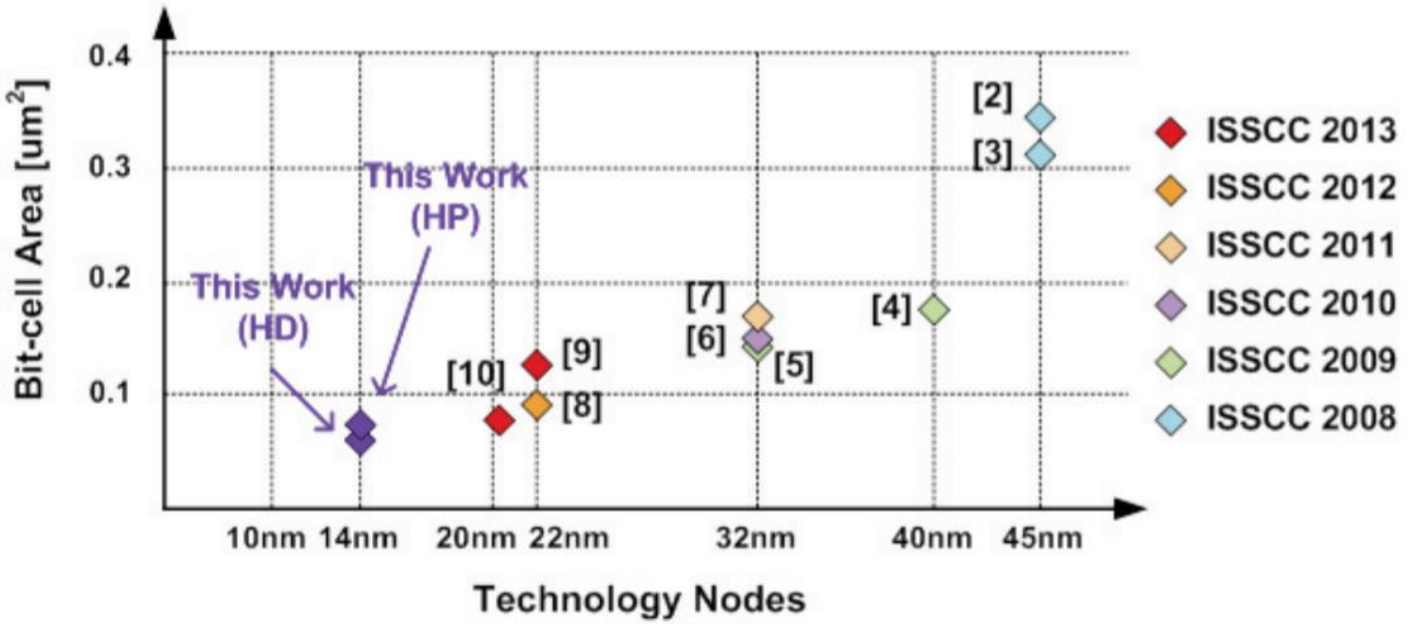
- Real memory technologies, area per bit
  - SRAM: 150-300 F^2
  - DRAM: $6F^2$
  - NOR FLASH $10F^2$
  - 2D NAND FLASH $4F^2$
  - 3D NAND FLASH $4F^2/n$ (n is number of layers)



Example: SRAM footprint



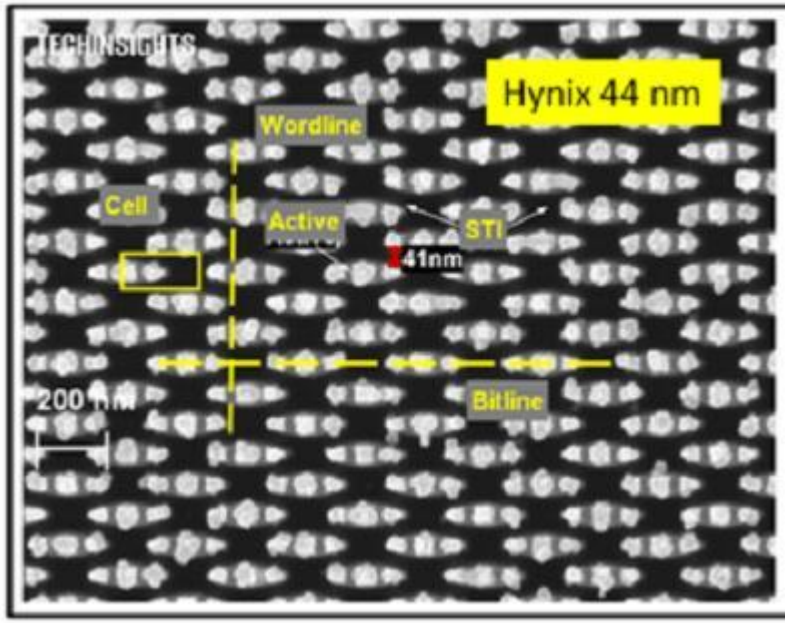
SRAM cell in the 14 nm node (Samsung)
$6F \times 13F = 78F^2$

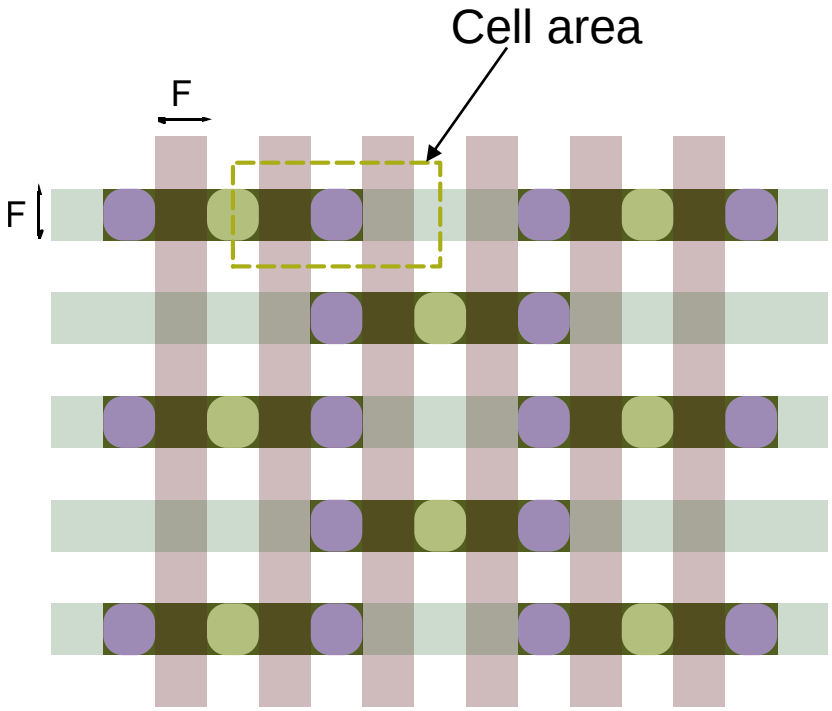


Song et al. (Samsung) ISSCC 2014

2 min exercise - Footprint

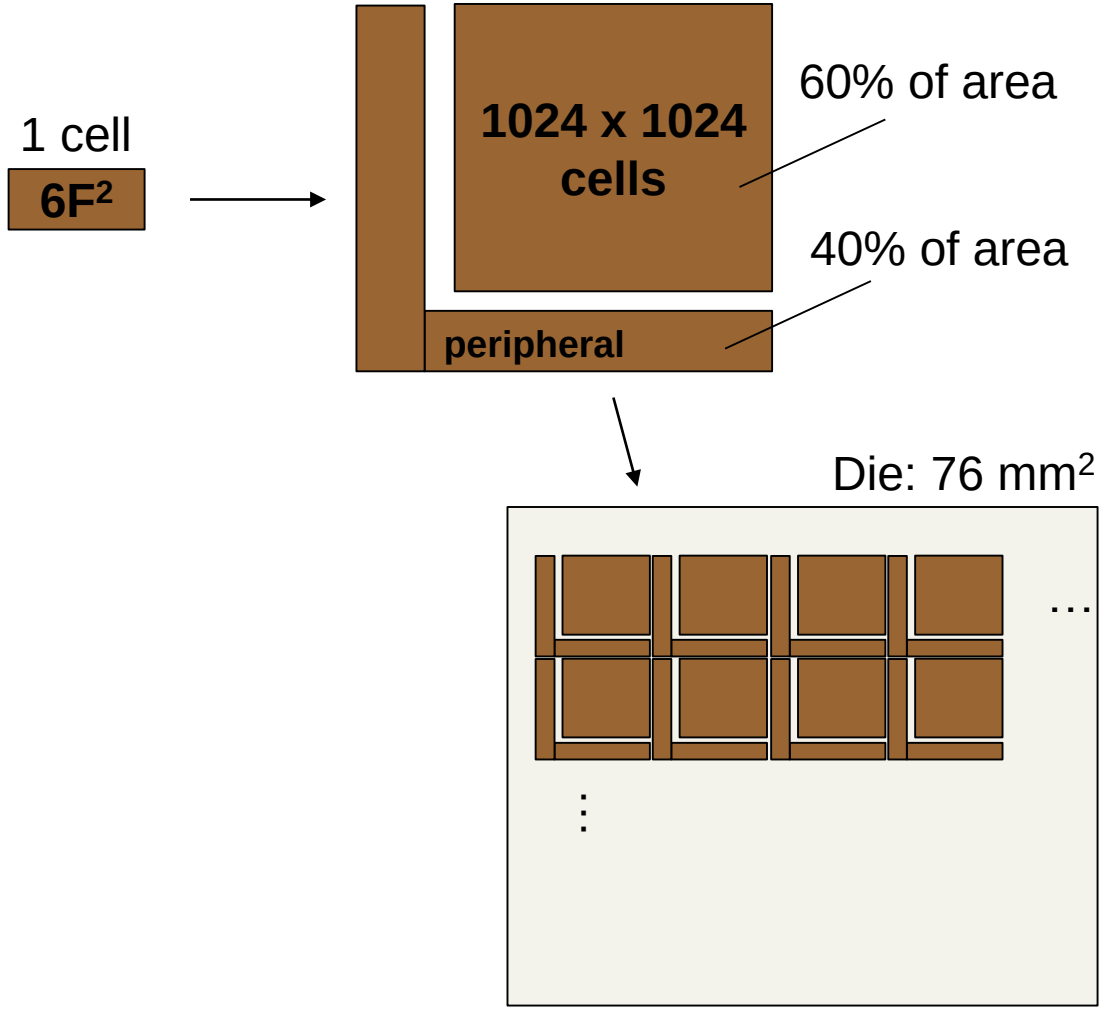
- Calculate footprint of a circuit (Hynix 44 nm DRAM). How large is the cell area?





Exercise 2: How much fits?

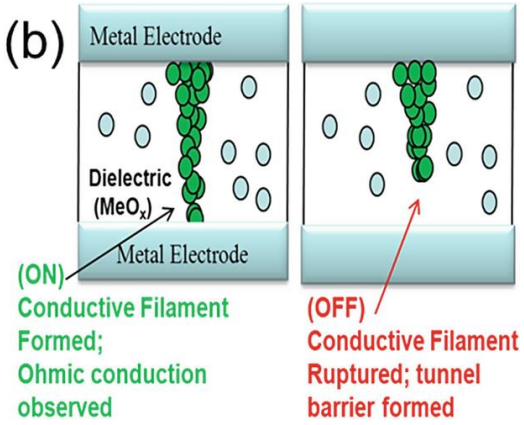
- How much DRAM fits on a die?
- Assume 1024 x 1024 memory cells in array
- Cell area = $6F^2$ *(best case scenario)*
- Area efficiency 45 %
- Die size 76 mm^2
- F = 26.7 nm



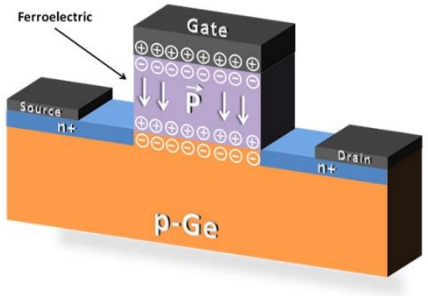
Opportunity for emerging memory devices

- Integrated on chip for fast access
- Non-volatile → energy-efficient
- Small latency (fast)
- Small footprint → high area efficiency
- Can we do all at once?
- Exchange for current memory technologies: SRAM, DRAM, Flash

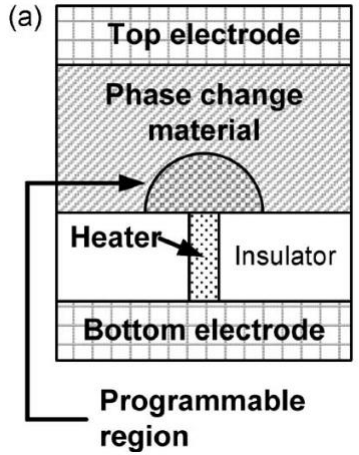
Conductive filament



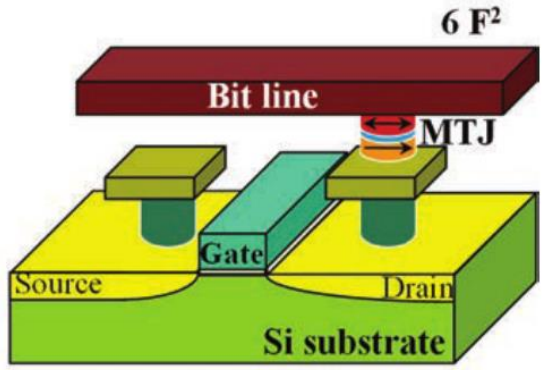
Ferroelectric



Phase Change



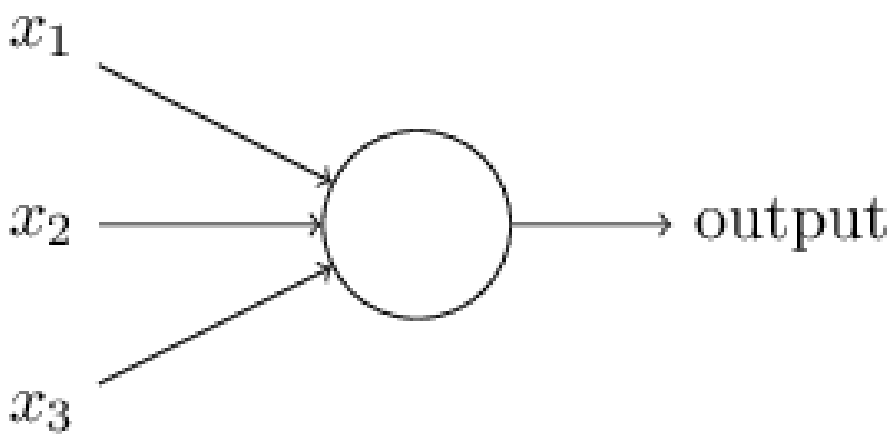
Magnetic



What about machine learning then?

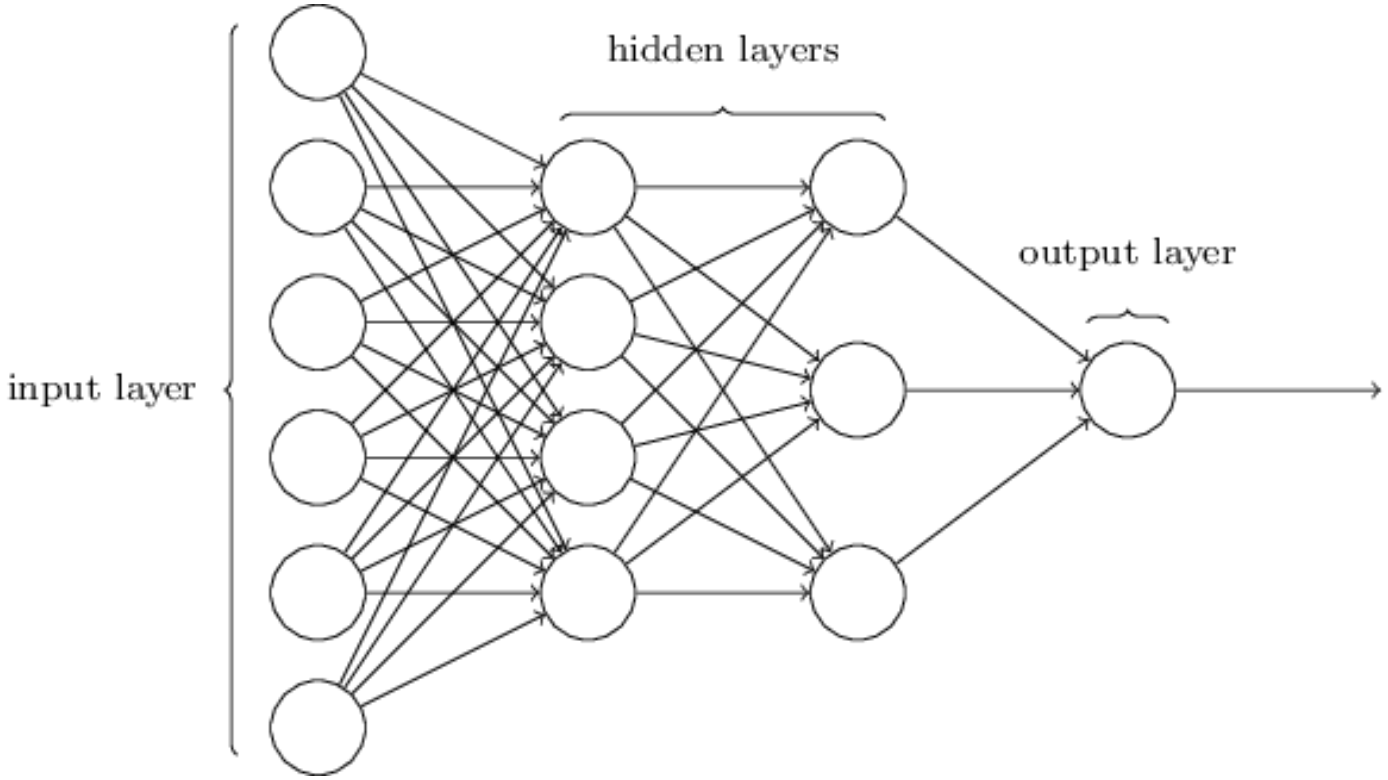
The Artificial Neural network I

The Perceptron Neuron Model:



$$\text{output} = \begin{cases} 0 & \text{if } \sum_j w_j x_j \leq \text{ threshold} \\ 1 & \text{if } \sum_j w_j x_j > \text{ threshold} \end{cases}$$

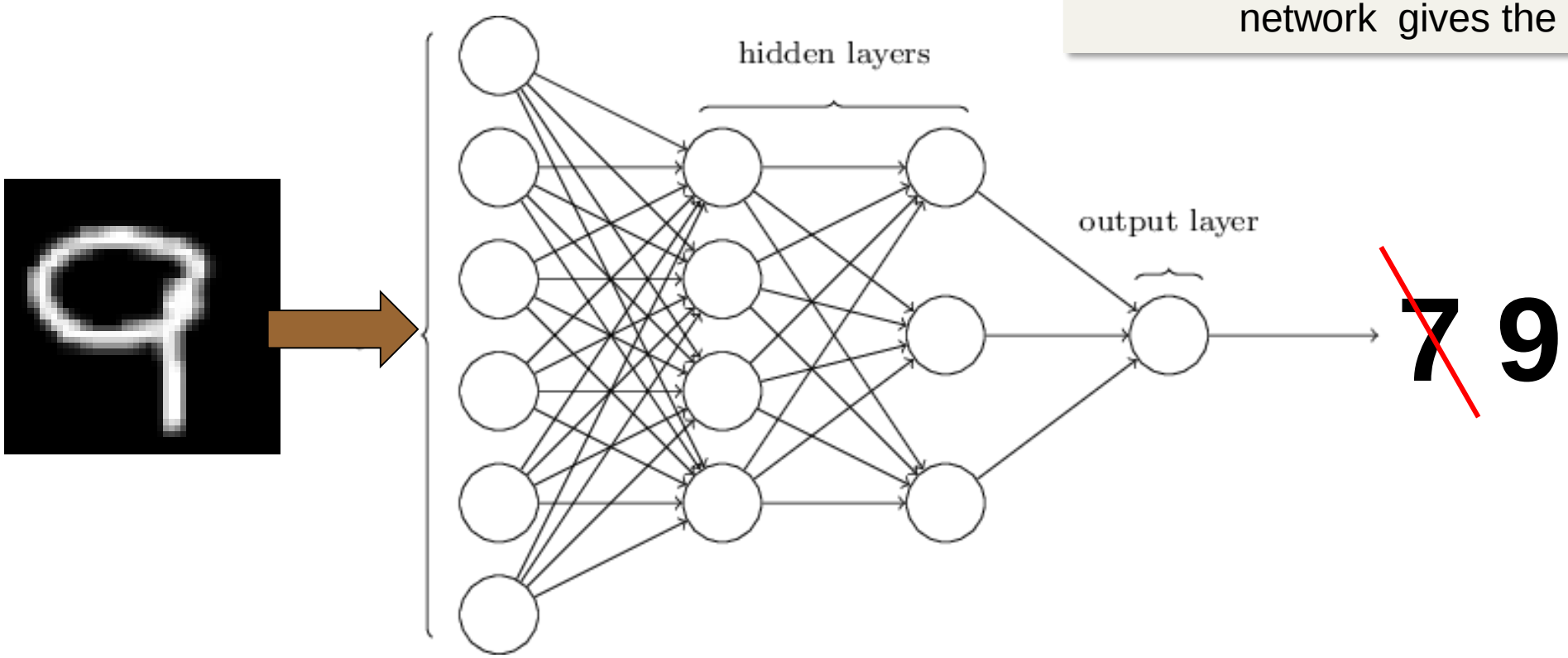
The Artificial Neural network II



How it works (on the surface)

Principle of Machine learning

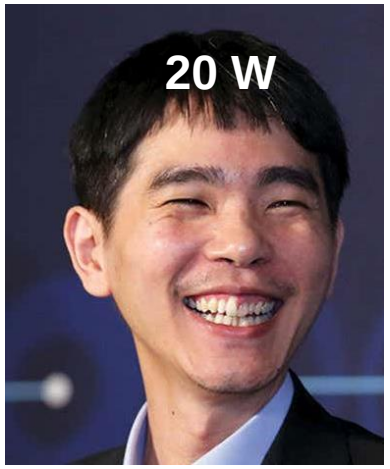
1. Push labeled data through network (*matrix operations*)
2. Check if outcome fits expectations
3. Adjust weights and biases in a smart way (*matrix operations*)
4. Repeat 1-3 with (tons of) more data until network gives the expected answer



The power problem of Machine learning

- March 19 2016: Google's Alpha Go beat Go Master Lee Sedol

- But:



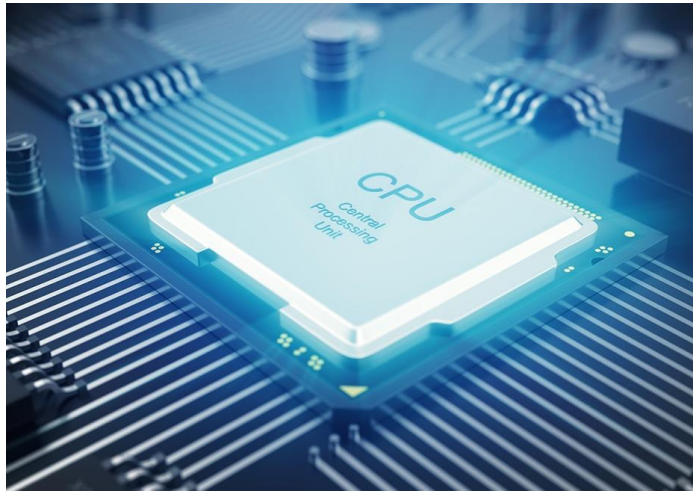
1920 CPUs + 280 GPUs
→ 1 MW (!!)*

*https://jacquesmattheij.com/another-way-of-looking-at-lee-sedol-vs-alphago/

Power problem in machine learning

Machine Learning is basically a matrix multiplication/inversion problem

General purpose hardware (CPU)
200 W, lots of memory but slow access

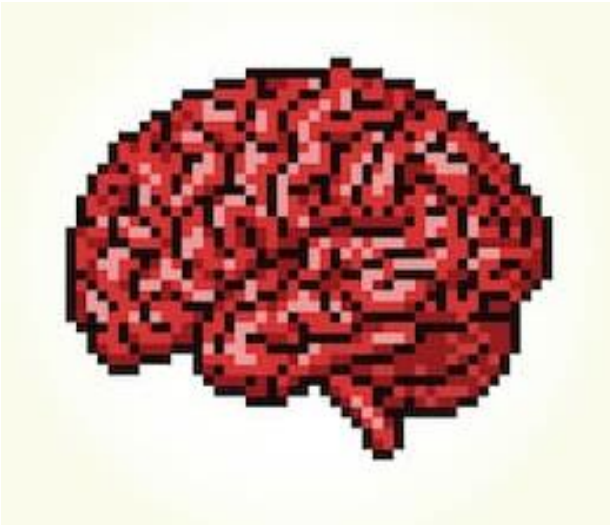


Specialized hardware for matrices (GPU, TPU)

250 W @ 100 teraflops

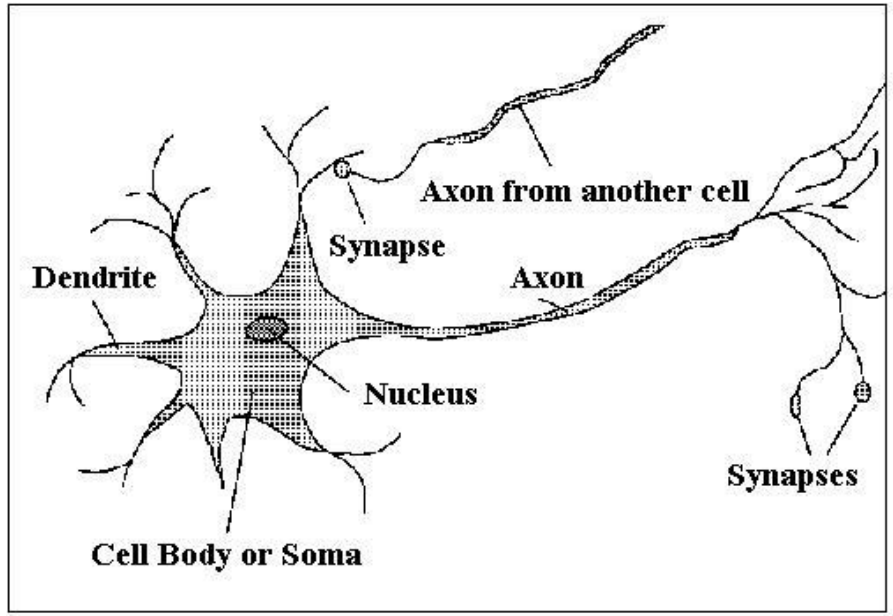
Human Brain

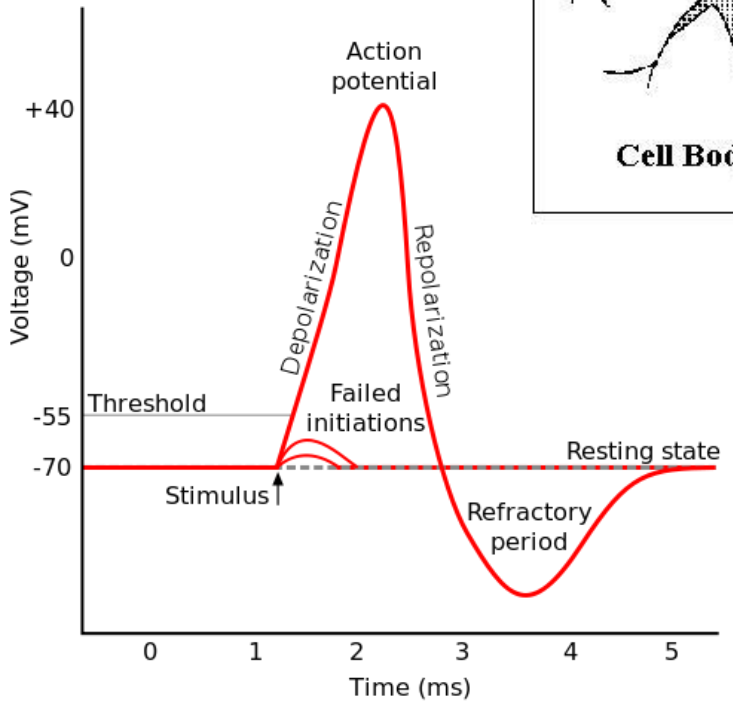
20 W @ 10^6 teraflops



How the brain works (”much simplified”)

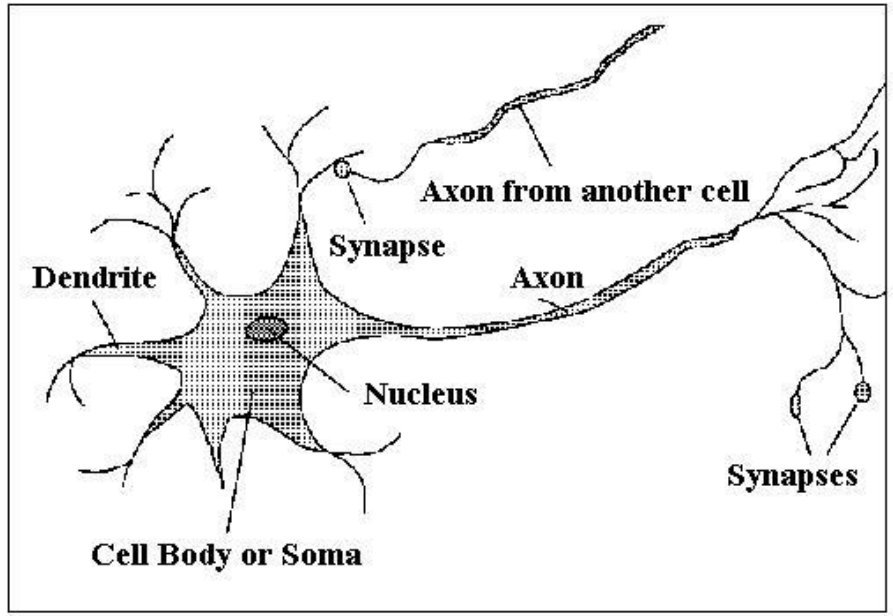
- Neurons ~ 10^{11}
- Connected by synapses with varying ”resistance”/”weight”
  ~ 10^{15} synapses
- Electrical stimuli above threshold close in time causes them to fire
- Signals propagate through network
- Connections encode logic/memory
- Hierarcial ”layered” structure

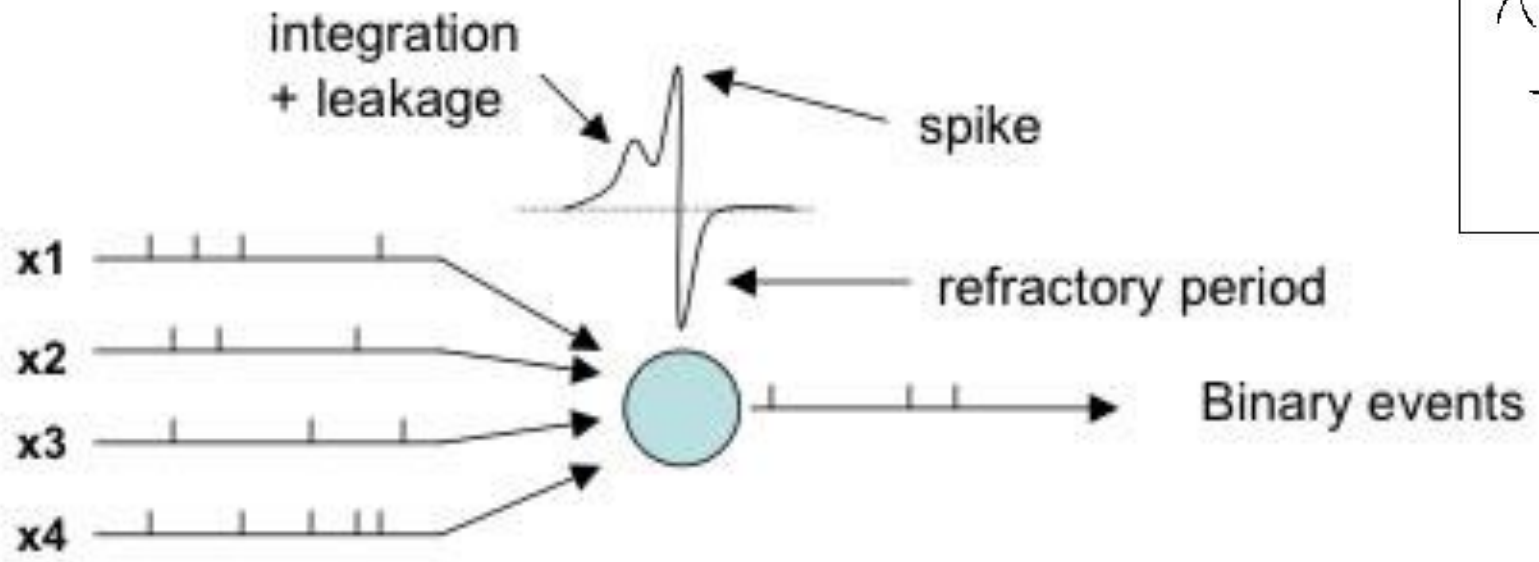




Spiking neurons

- Input signals in dendrites are integrated in the neuron
- Many inputs in short time interval
  → a threshold is overcome
  → the neuron will fire a signal into the axon

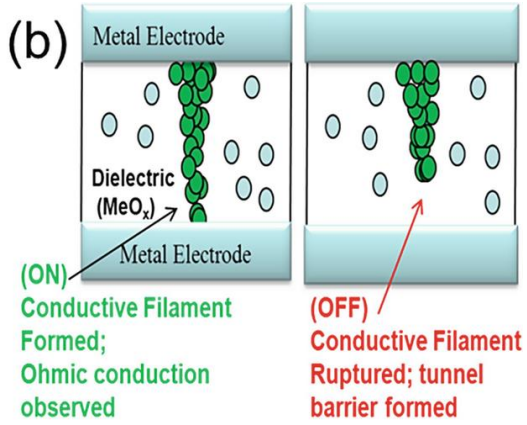




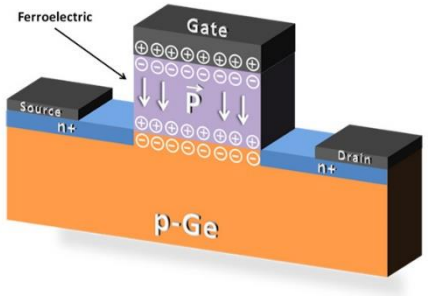
Memories in Neuromorphic Computing

- Emerging non-volatile memories
  - Resistance-based! → ”weights”
- Can we use memories to make synapses and neurons in hardware?

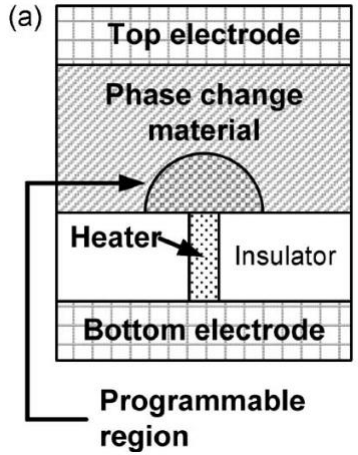
Conductive filament



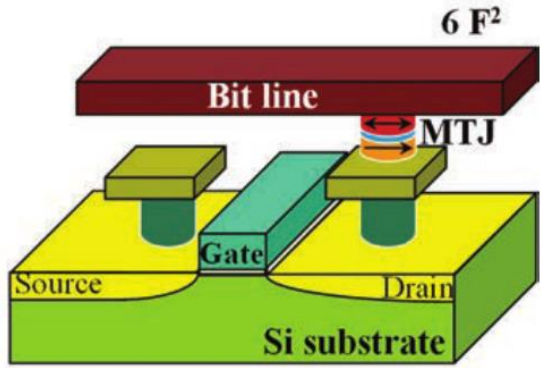
Ferroelectric



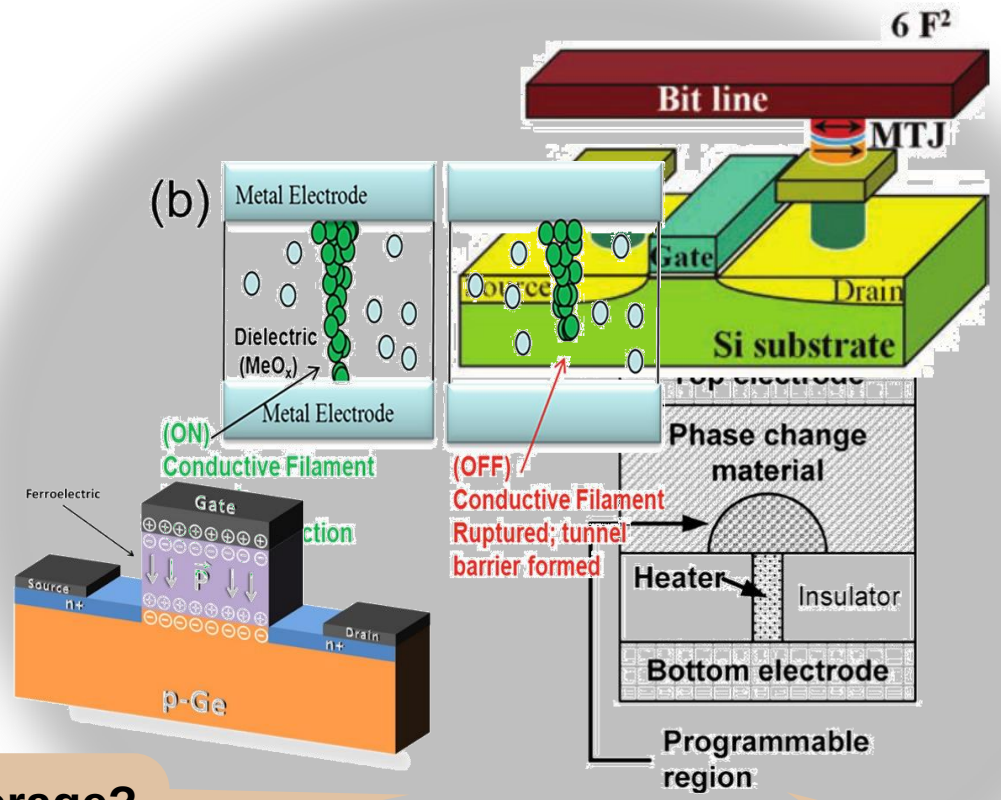
Phase Change



Magnetic



Summary



As computational units for in-memory computing?
Accelerating machine learning

As efficient memory for storage?
Replacing conventional memory technologies

As building blocks for neuromorphic computing?
Synapses and neuron devices