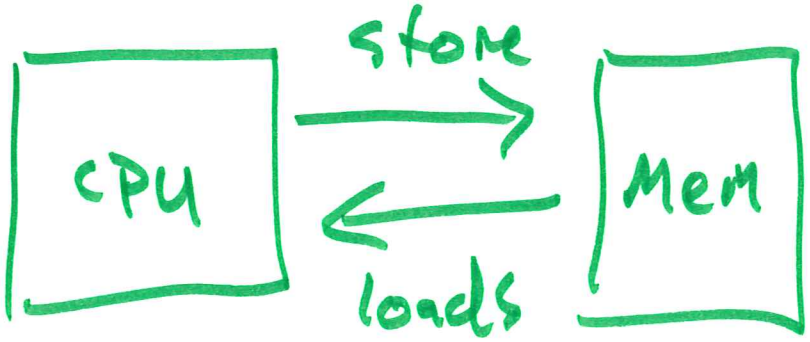
The Memory System

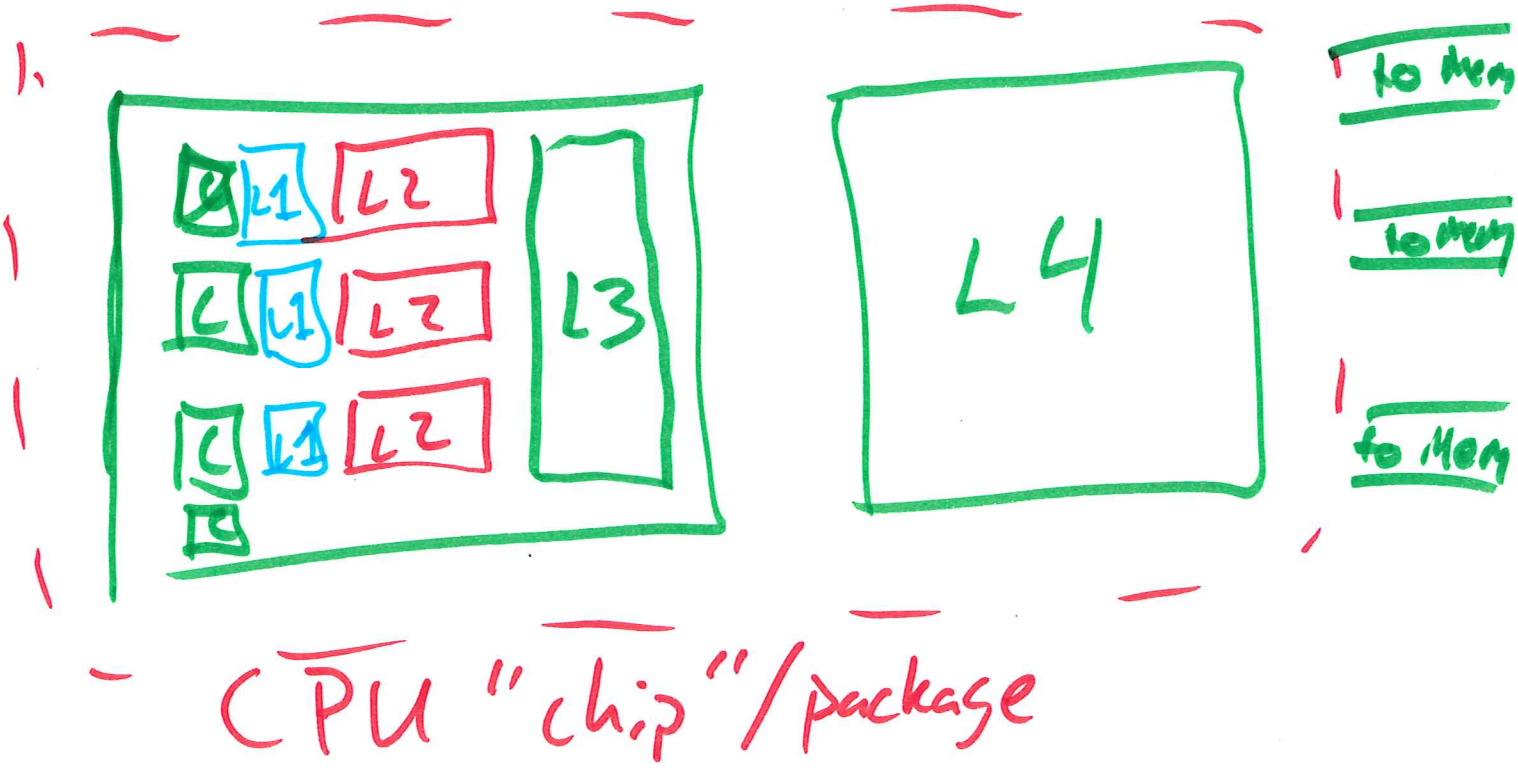
load + store instructions

movl in x86

20-50% of all instructions



But it's actually much more complicated

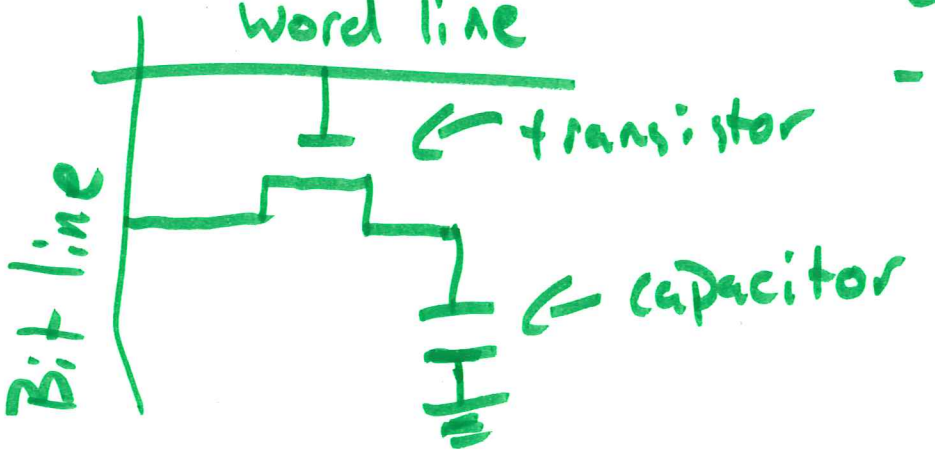


RAM: Random access memory

DRAM -> Dynamic (must be refreshed)

SRAM -> Static RAM

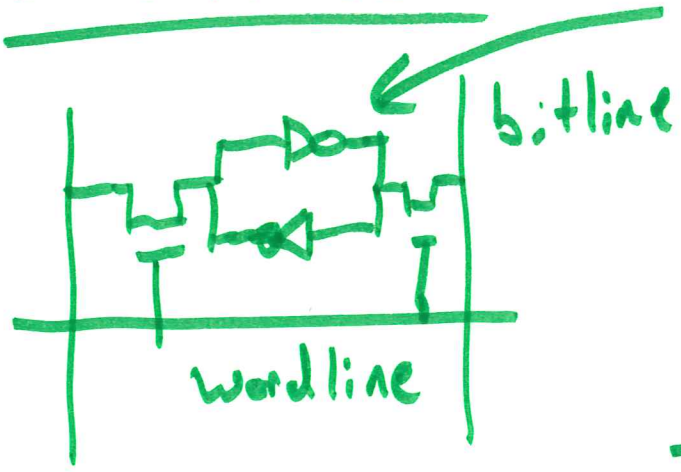
DRAM cell



- 1 if capacitor is 0, otherwise it Charged
- Capacitor leaks over time
  - ↳ refresh the charge
  - 1 - 100 ms

- DRAM cells are <u>very</u> small
- Goal of DRAM is <u>highest density + lowest cost</u>

SRAM cell



2 not gates
in a feedback loop

- 6 transistor cell
  some cells up to 12t
- no need to refresh
  - but still leakage

- Goal with SRAM → speed or low power
  - speed → L1 cache or registers
  - low power → L2 L3 cache

Storage devices

disks → still very common

non-volatile memories primarily flash (but soon others?)

Disks → spinning platters

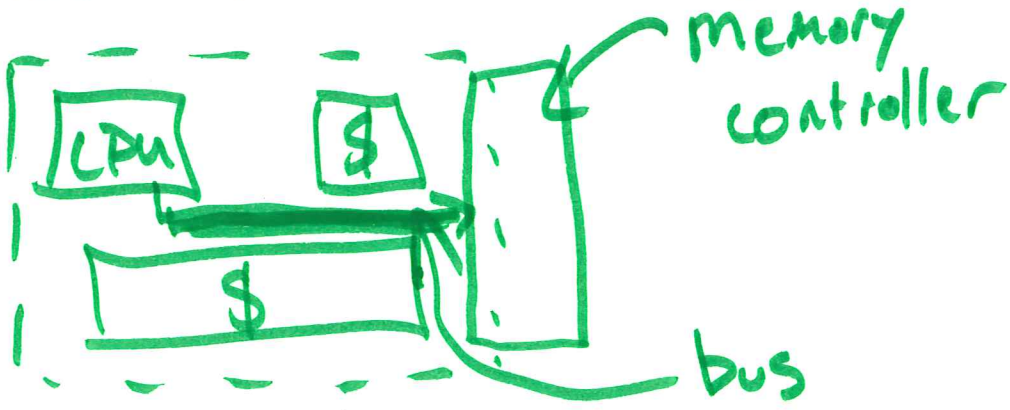
- arm with a head that
  - detect magnetic variations

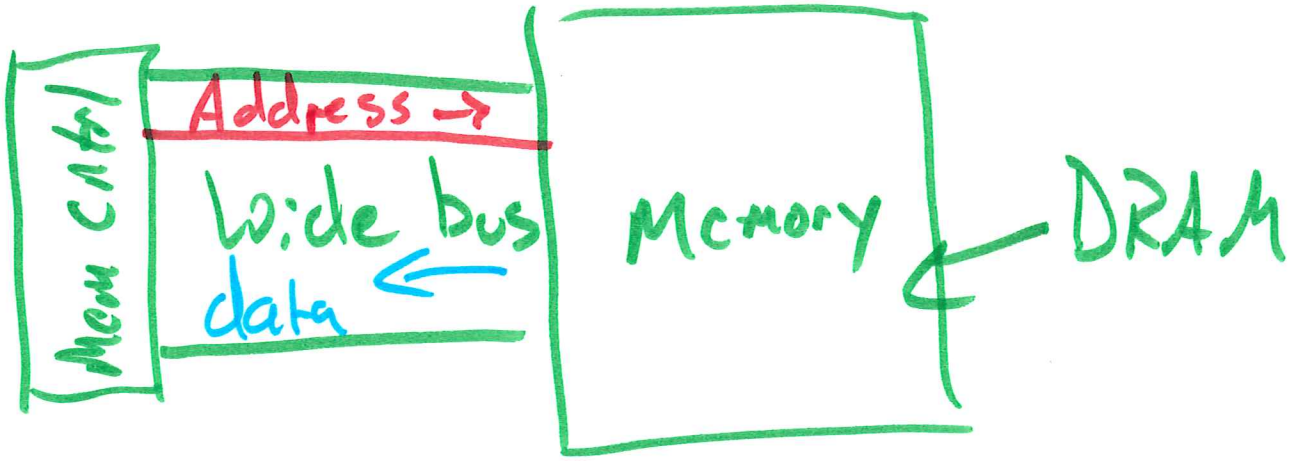
- Many more details later

Flash - "floating gate transistor"

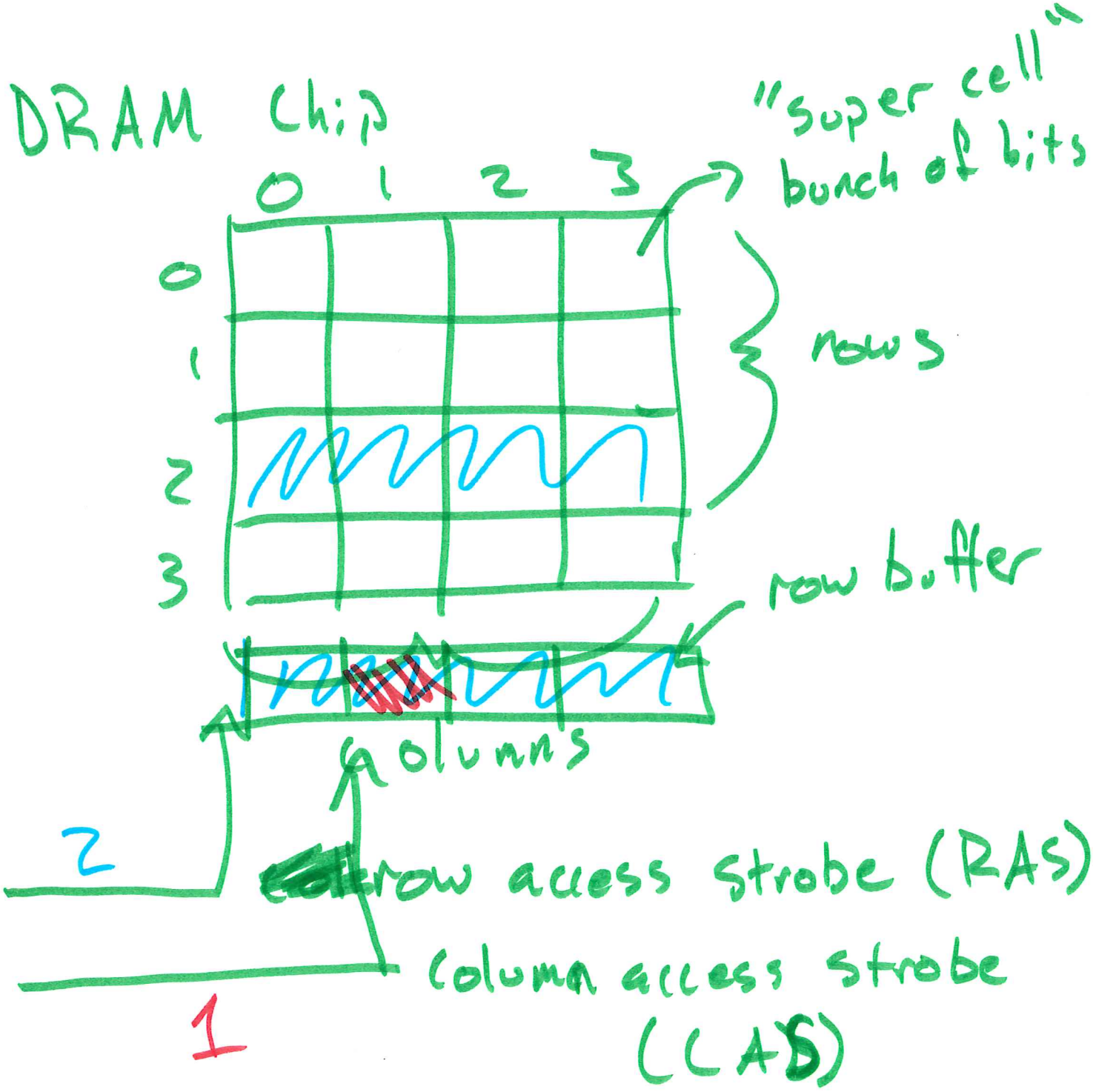
- kind of like DRAM, but <u>not</u> a capacitor
- faster than disk
  slower than DRAM
- Much denser than DRAM
- Many drawbacks (limited # writes)
  (read in blocks)

More DRAM details




Mem ctrl
Address →
Wide bus
data
Memory
DRAM

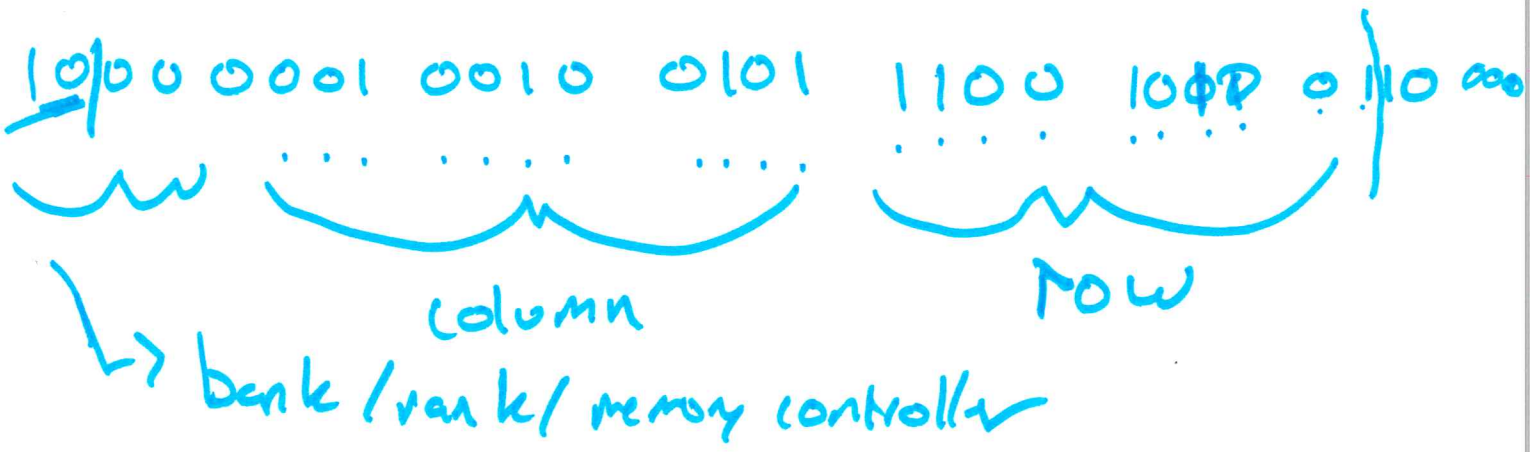
DRAM Chip


"super cell"
bunch of bits
0 1 2 3
0
1
2
3
rows
row buffer
columns
2
row access strobe (RAS)
1
Column access strobe
(CAS)

Memory controller gets an address

0x8125ca70

going to get 64 bytes → ignore bottom 6 bits



DDR2 →

→

DRAM DIMM → Dual inline memory module

DDR3 → common today

DDR4 → becoming common

GDDR5 → graphics DRAM

LPDDR 2/3/4 → phones

Cool new memory technologies

HBM -> high bandwidth memory

HMC -> hybrid memory cube

Wide IO