

lec 17

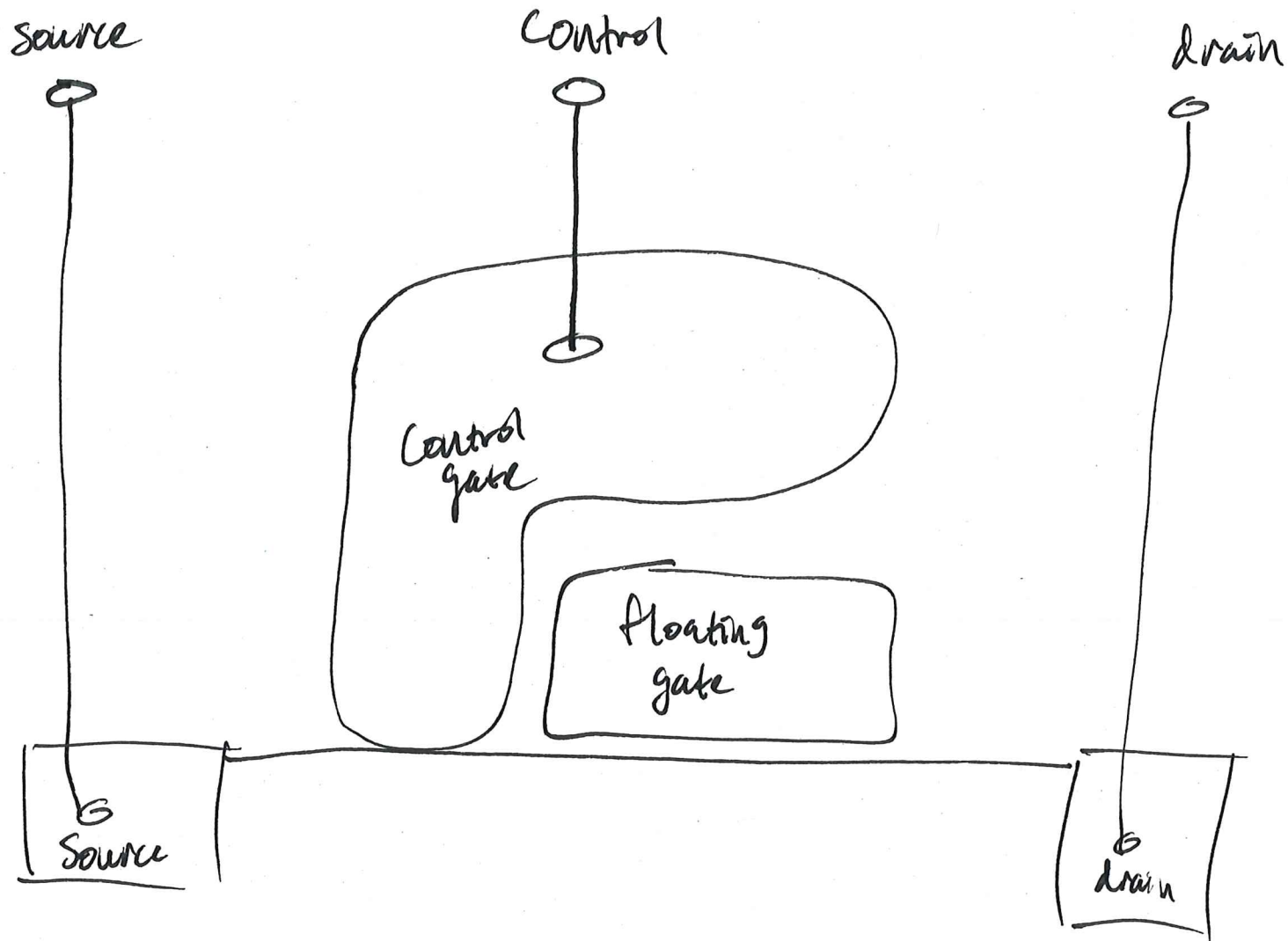
Storage

Volatile Memory

- data is lost when power turns off
- word (8 bytes) or cache line (64 bytes) access
- Static RAM (SRAM): registers, on chip caches
 - transistor-based flip-flops store the data
 - latency: 1-20 ns
- dynamic RAM (DRAM): main memory (separate chip!)
 - data stored in a capacitor
 - capacitors leak; requires "active refresh"
 - latency: 50-100 ns

Persistent Memory

- data is valid even without power
- block-level (4Kb or more) access; async requests
- NAND flash / solid state drive (SSD)
 - bits stored in floating gate transistor
 - latency: 10-50 μ s block-level random read/write
- Magnetic "hard" disk drive (HDD)
 - bits stored on magnetic surface
 - physical motion (spinning) required to access bits
 - 10 ms random access latency



Flash memory

17.3

Flash Challenges

- can only write to "clean" cells
 - must erase cell to clean it
 - erase is 20x slower than write
 - requires guard regions to protect neighbors
 - gradually wears out the cell
- ↳ can only erase "erasure blocks", ~256 Kb

Managing Wearout

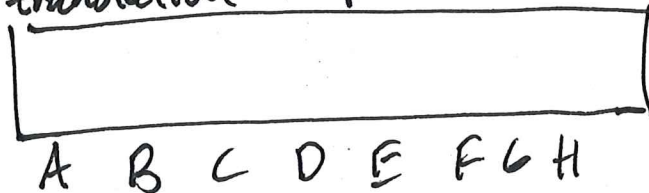
- each block can be erased only limited # of times
 - 10k typical for multi level flash
- What if system repeatedly updates one phys block
 - 1ms write/erase cycle $\rightarrow$ 10s to wearout!
- Can't allow update in place
 - (also if system wants to update just one page then whole erase block would be rewritten)

Flash Translation Layer (FTL)

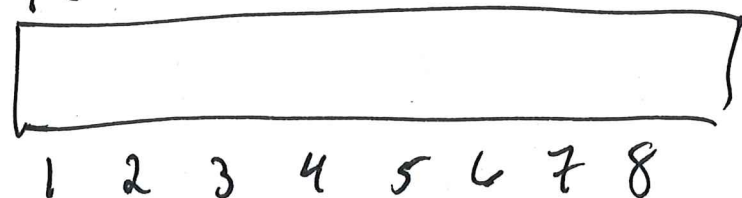
- firmware presents virtual block interface
 - block-level read/write/trim (free)
- internally maps each virtual block to physical
- overwrite virtual block → write to clean block + remap
 - previous physical location is stale/unused/free

Example: writes C E A D C B A

translation map



physical storage



Erasure

- eventually must reclaim stale blocks
- not all pages in an erasure block might be stale
 - ↳ relocate live pages elsewhere
- good candidates for erasure:
 - erasure blocks w/ few live pages
- reserve 20-30% of space for flexibility
 - delay compaction → increase chance of few relocations

Wear Leveling

- FTL tracks # of erasures per block
 - stop using block if it is worn out
- avoid wear out by writing to least used blocks
- put long-lived data in more worn blocks.