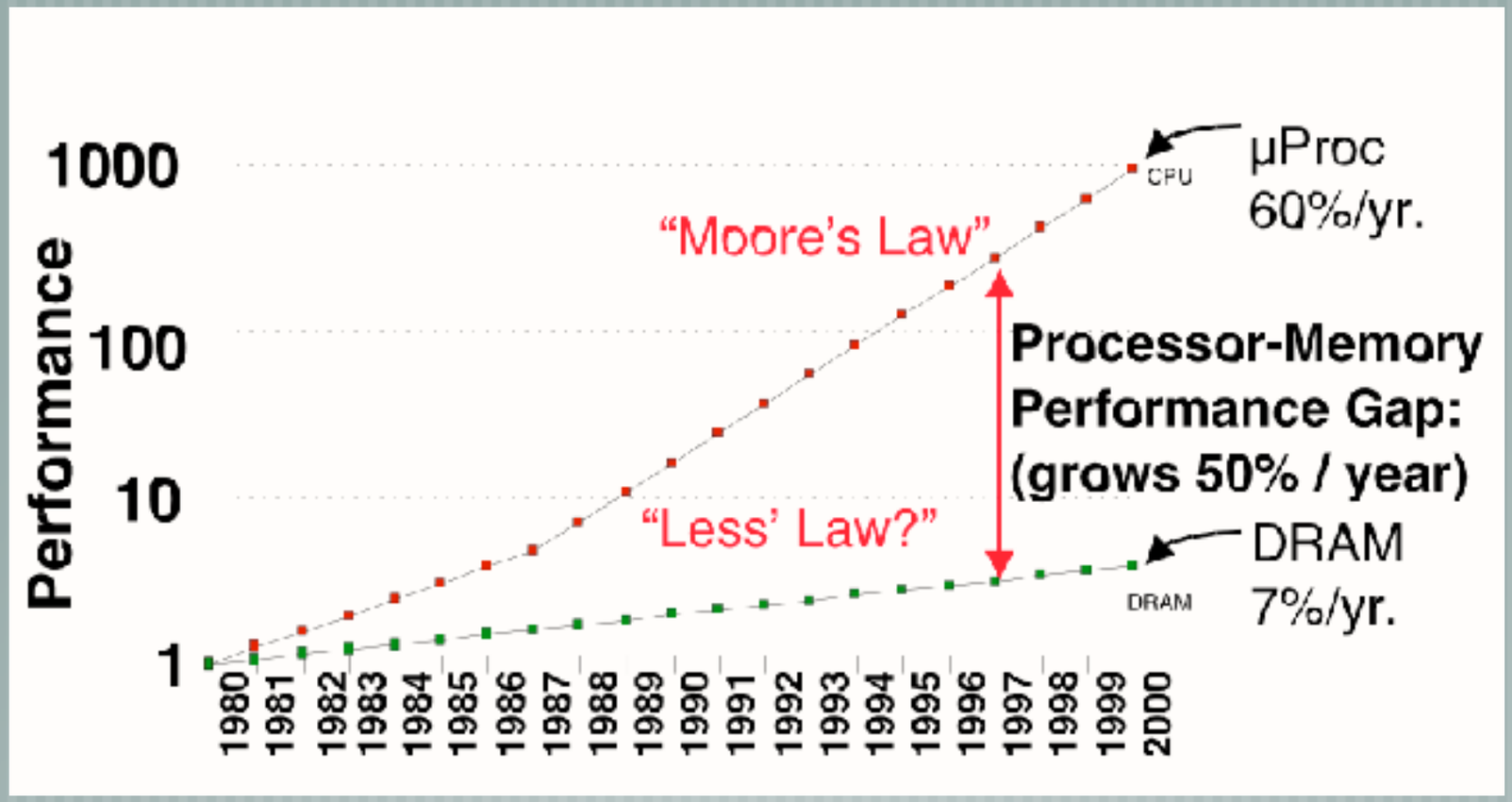
1. Memory technology & Hierarchy

RAM types

Parallel System Architectures
Andy D. Pimentel

Memory wall



- "Memory wall" = divergence between CPU and RAM speed
- We can increase bandwidth by introducing concurrency in memory access (e.g. through pipelining accesses)
  - but this requires regular access patterns
  - random accesses to main memory can cause severe performance degradation

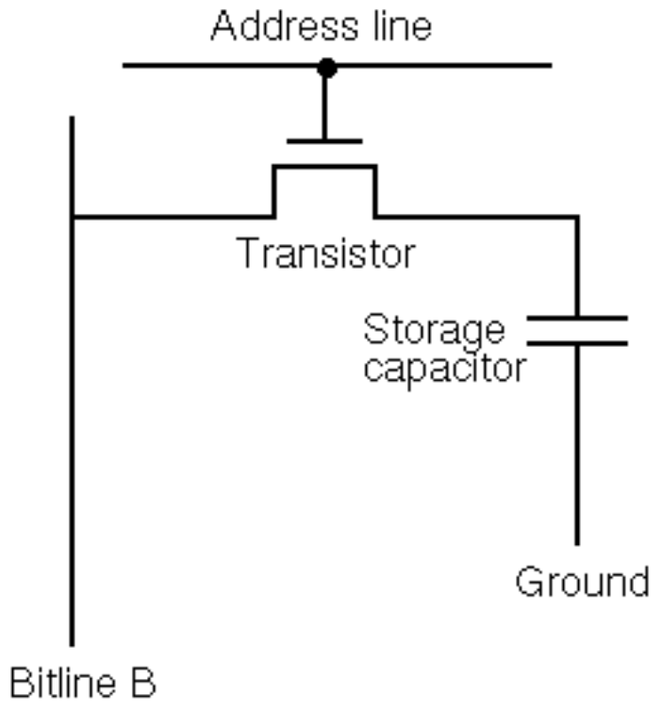
Memory hierarchy - issues

- Conflicting requirements in a memory system: we want both **large** and **fast**
- Electronic systems have higher latencies as they increase in size
  - **Speed of light is approximately 1ns for 30cms**
    - N.b. 1 ns is 3 clock cycles in a state of the art processor
  - **Pins, wires, connectors etc. all add resistance and capacitance which delay signals significantly**
- A memory hierarchy attempts to make a large slow memory appear fast by buffering data in smaller faster memories close to the processor

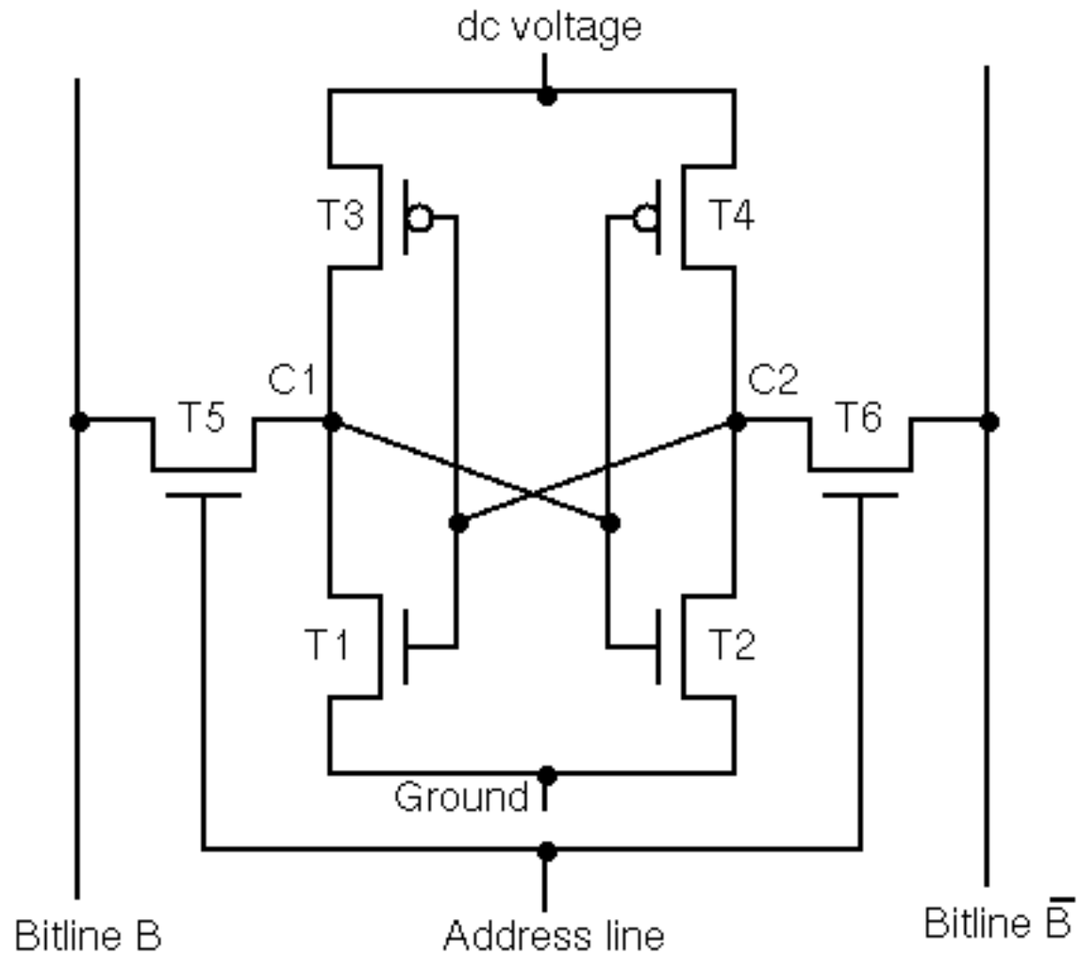
Memory hierarchy - issues

- Can make memories faster but this requires more power
- Need to drive long wires on chip across memory array – to do this fast needs more power
- Memory performance is a compromise between power and performance
- (As is processor performance today)

RAM cell designs



DRAM cell

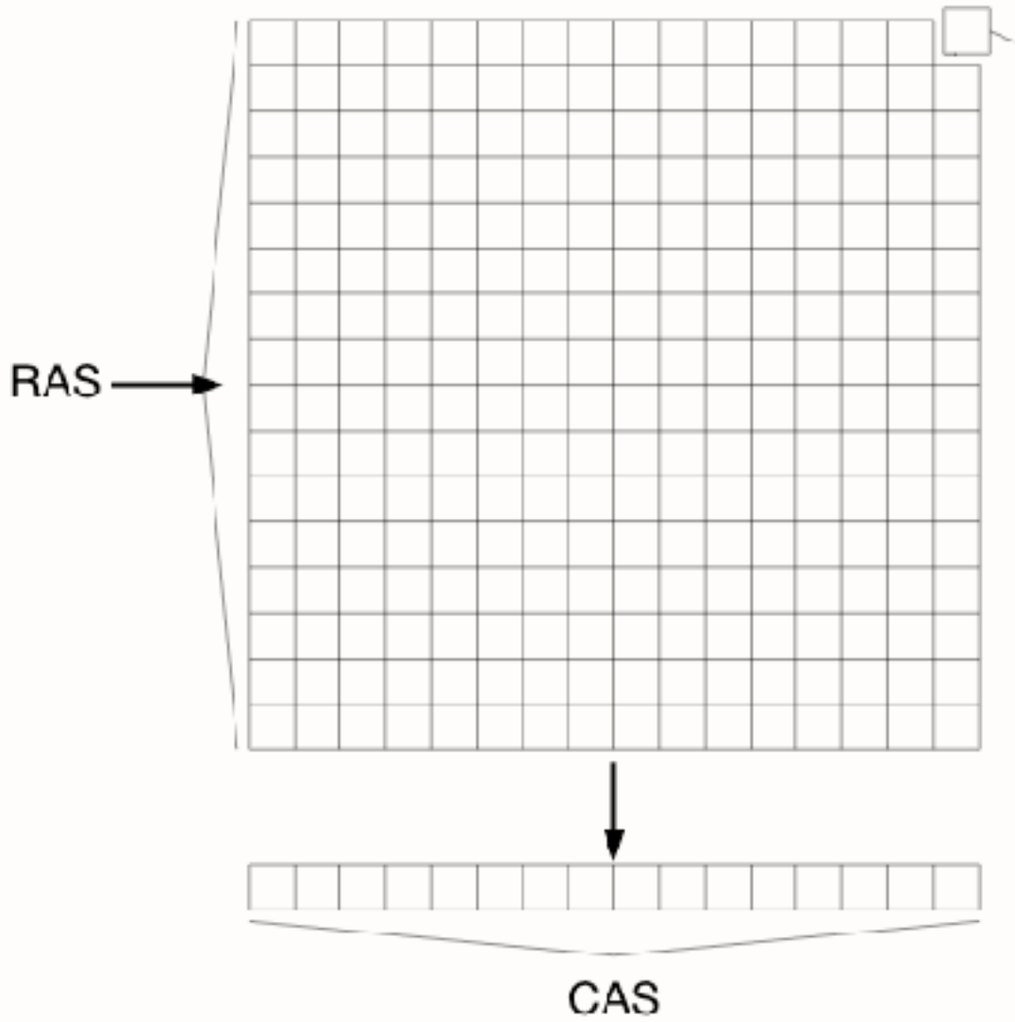


SRAM cell

Components - DRAM

- DRAM - is a very **dense** form of RAM - it is volatile
  - **access is destructive & data must be re-written**
  - **charge also leaks from the capacitor which stores the data so data must be refreshed periodically (“dynamic” RAM)**
- Typical DRAM chip characteristics
  - **256-1024 Mbit and 2-800MHz cycle**
- Uses two cycle Row/Column addressing (RAS/CAS)
  - **two-stage access and requirement to rewrite data contribute to slow memory access times**
  - **but good design can help for regular accesses**

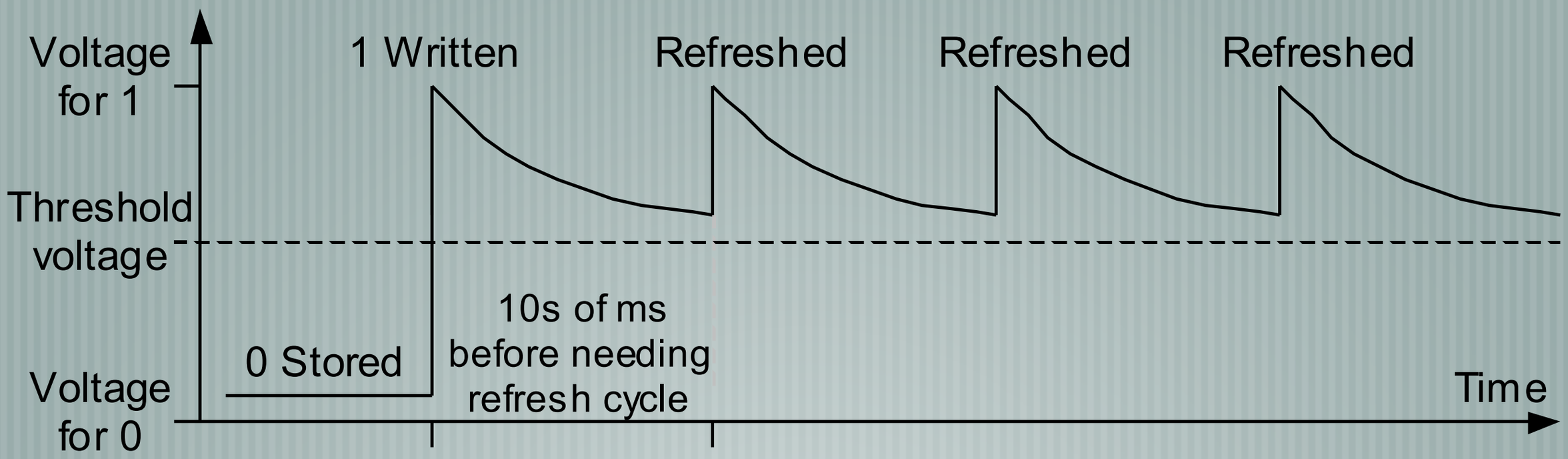
DRAM - RAS/CAS addressing



Step 1: Row Address Select
Step 2: Column Address Select (select bit)

Refresh: read and write back a whole row

DRAM refresh



Components - SRAM

- SRAM - is less dense but faster than DRAM
  - **uses four transistors to store data and two transistors to access it - access is non-destructive**
  - **data is stable while the RAM has power connected ("static" RAM)**
- Typical SRAM chip characteristics
  - **~64Mbit and 100MHz-1GHz cycle**
    - although SRAM cycle times are similar to DRAM, SRAM is true random access memory
      DRAM can only read consecutive bits at the cycle rate, therefore DRAM has a much larger latency time
  - SRAM is also used for memory on the processor chip
    - **registers** and **cache** both use SRAM technology
- **the smaller the memory the "faster" it operates: lower latency**

Bandwidth vs. Latency

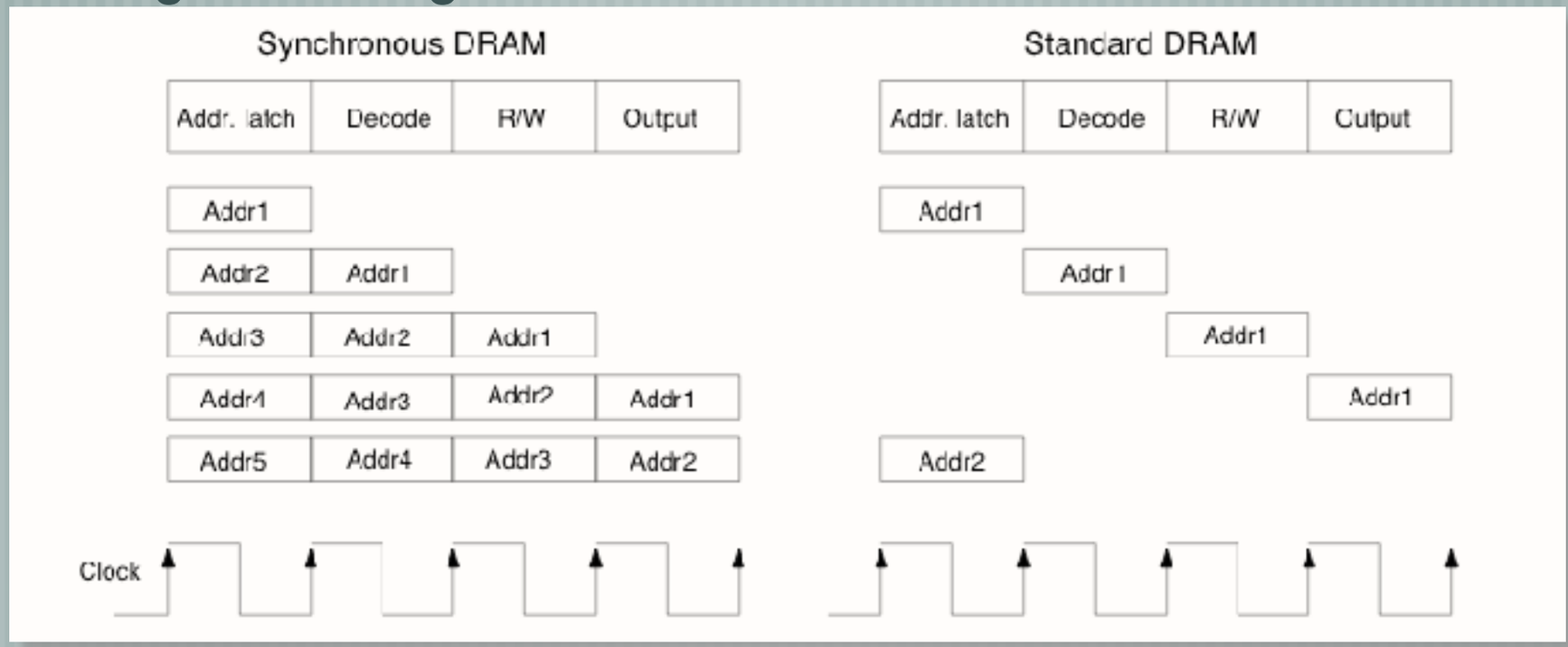
- **Memory latency** is the time delay required to obtain a specific item of data
- This is larger in DRAM than in SRAM
  - SRAM can access any bit each cycle
  - DRAM is restricted to bits in the same row, CAS cycles
- **Memory Bandwidth** is the rate at which data can be accessed (e.g. bits per second)
  - **Bandwidth unit is normally 1/cycle time**
  - **This rate can be improved by concurrent access**

Improving DRAM bandwidth

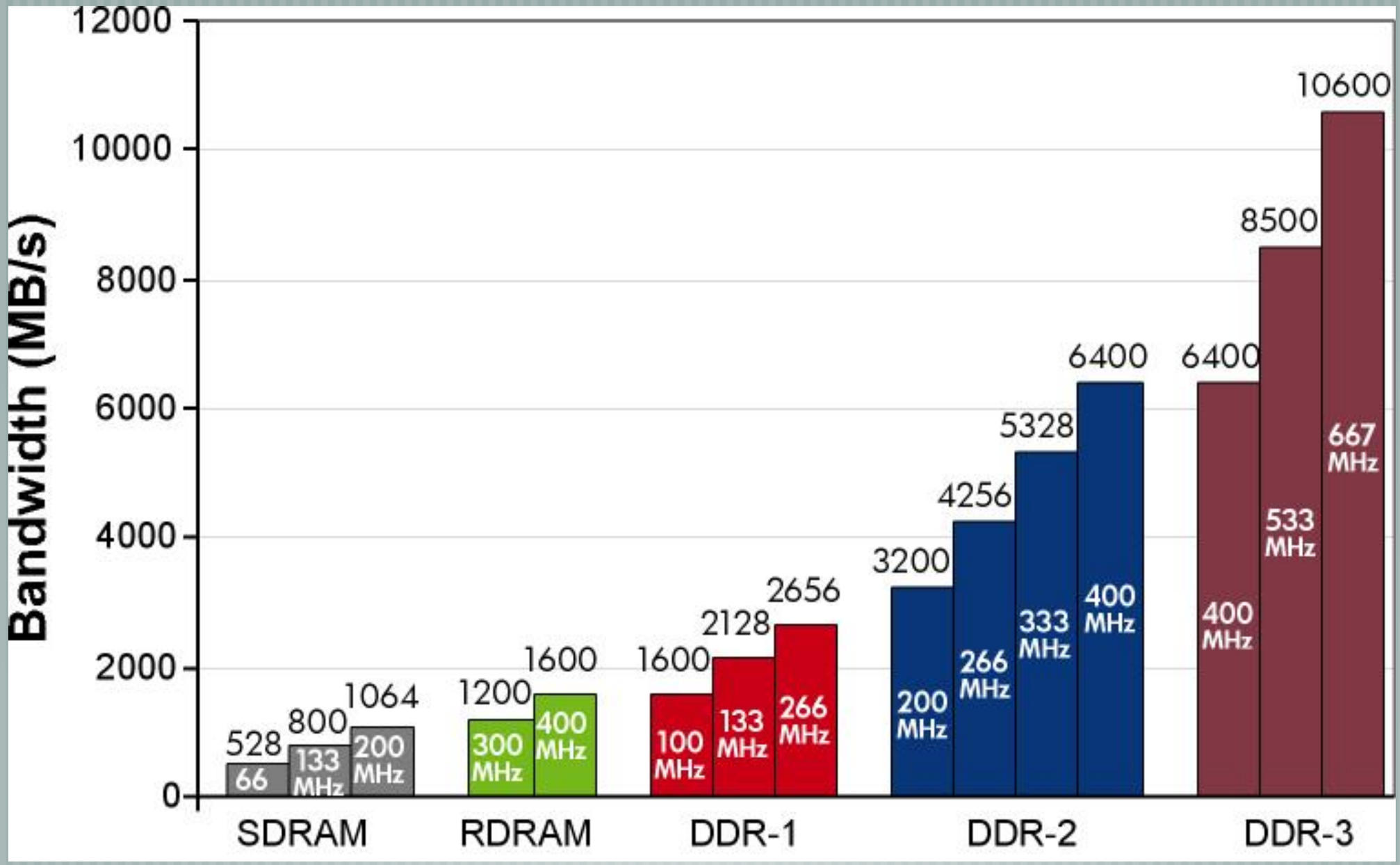
- Using locality to get maximum bandwidth
  - **One RAS multiple CAS e.g.**
    - Fast page mode DRAM
    - E(xtended) D(ata) O(utput) RAMs
  - **Burst-mode DRAMs**
    - Burst EDO RAM
    - S(ynchronous)DRAM
- By improving the interface
  - **DDR SDRAM and RAMBUS**

Example - Synchronous DRAM

- SDRAM changed the memory interface from asynchronous to synchronous and uses a form of pipelining – will return to this concept later
  - **DDR SDRAM** - double data rate uses transfers on both rising and falling clock edges



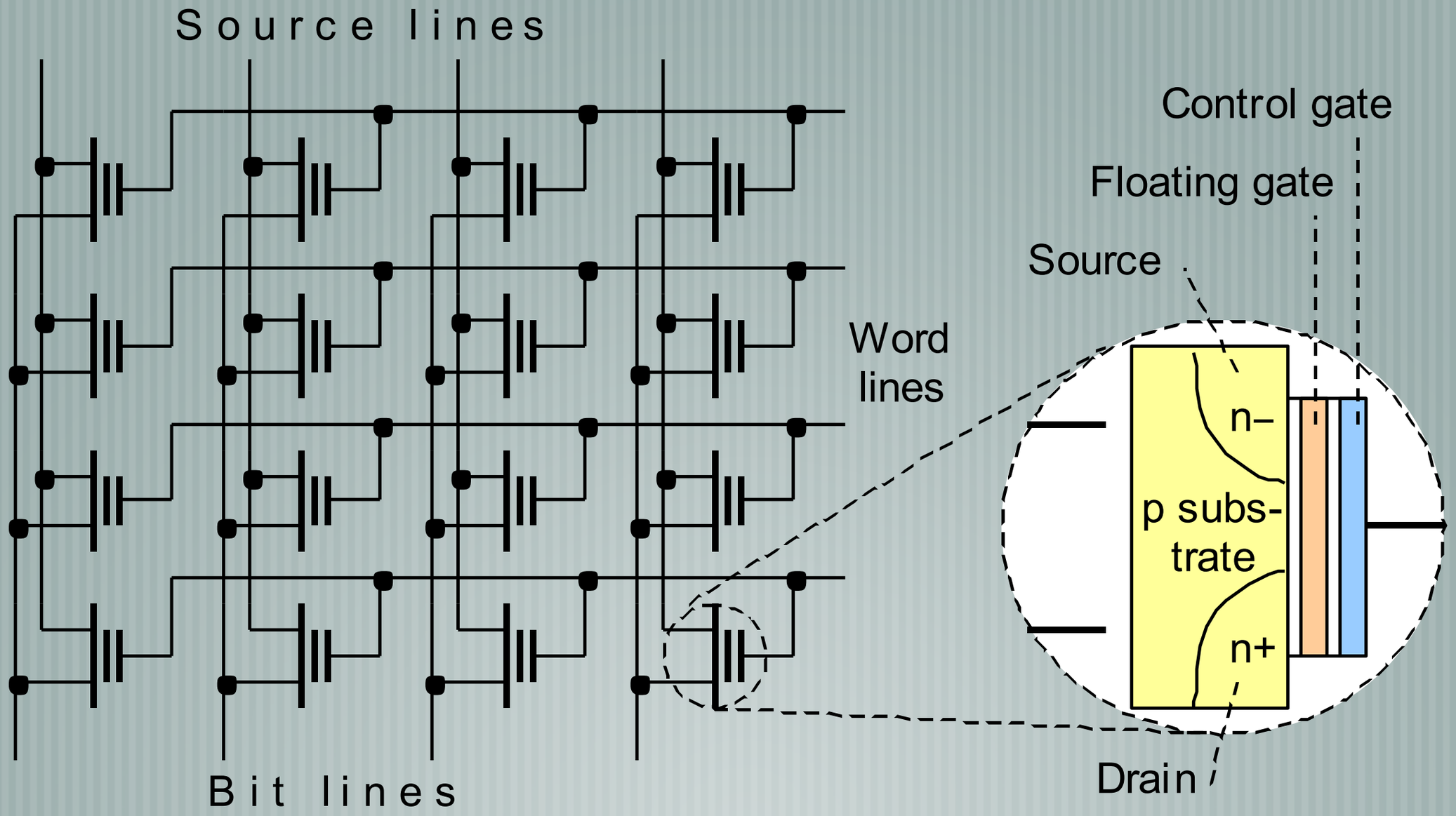
Different SDRAM technologies



RAMBUS DRAMs

- This is an interface improvement using a pipelined bus interface sometimes called a split-transaction
  - **Bus comprises row and column address line + 18 bits of data**
  - **3 transactions on bus simultaneously (RAS/CAS/Data)**
  - **High clock rate (400MHz) with data transfers on both edges**
- **Note that neither technique (SDRAM or RAMBUS) can improve the latency to access a single item of data**

Flash memory, non-volatile



A lot of ongoing memory research

- New memory technologies, mostly non-volatile, to replace DRAM
  - MRAM/STT-RAM - Magnetic
  - RRAM/Memristor - resistance is a function of the history of the current through the device
  - Phase-change memory (PCM) - change state of material (amorphous / crystalline) with different electrical properties
- 3D stacking of memory on top of logic (i.e., processor)

Memory wall – solutions

- The most common solution to the memory wall is to cache data
  - **requires locality of access or memory reuse**
  - **compiler optimisations can help to localise data**
- Can also design banked memory systems to provide high bandwidth to random memory locations
  - **Some access patterns will still break the memory**
- Can design processors that tolerate high-latency memory accesses "don't wait do something else"
  - **Requires concurrency in instruction execution**

Summary

- Need for new dense + fast technology
- Many bandwidth improvements, not so much for latency
- Most common solution: **cache** data
  - Requires locality of access, or memory reuse
- Design processors that **tolerate high-latency memory accesses** – "don't wait do something else"

Summary

Limits of RAM components

cause

Memory wall

problem

Caches
& hw multithreading

solutions

New components
(?)