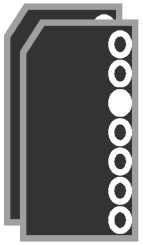
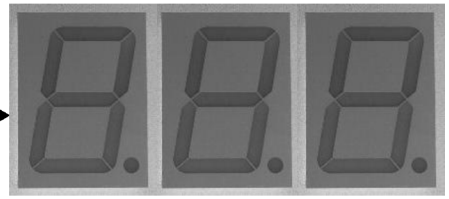
Memory

Kevin Walsh
CS 3410, Spring 2010
Computer Science
Cornell University

See: P&H Appendix C.8, C.9





3

mux

mux

LED dec

+1

32

32

32

32

reg E

reg E

reg E

reg E

decoder (3-to-8)

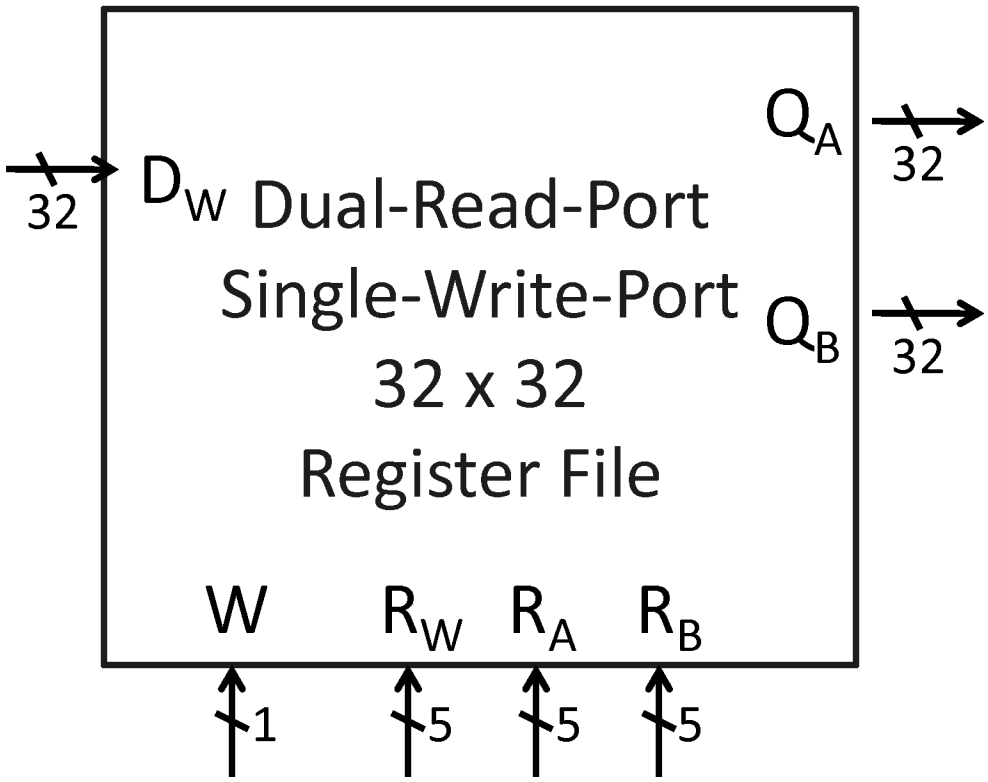
detect

enc

3

Register File

- N read/write registers
- Indexed by register number



Implementation:

- D flip flops to store bits
- Decoder for each write port
- Mux for each read port

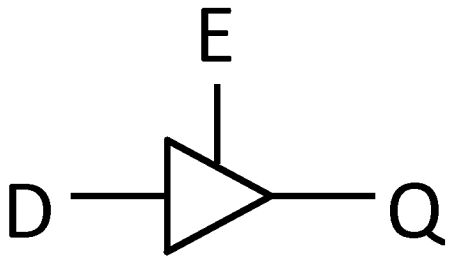
Register File tradeoffs

+ Very fast (a few gate delays for both read and write)
+ Adding extra ports is straightforward
– Doesn't scale

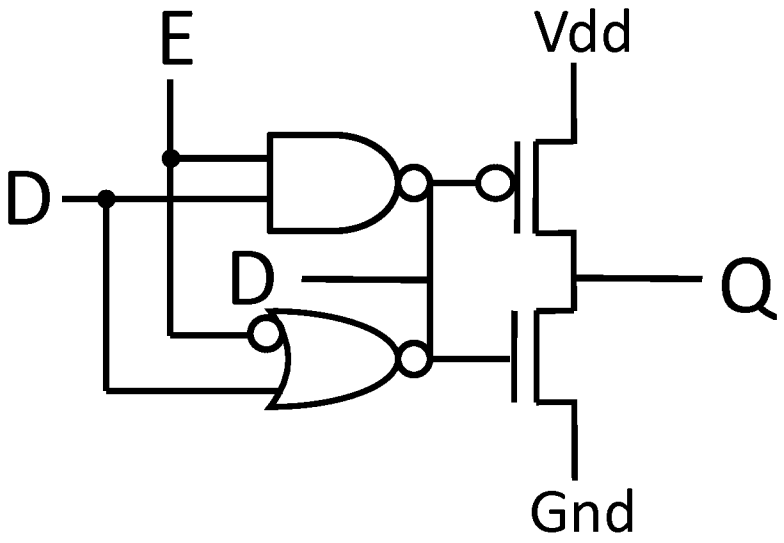
Need a shared bus (or shared bit line)

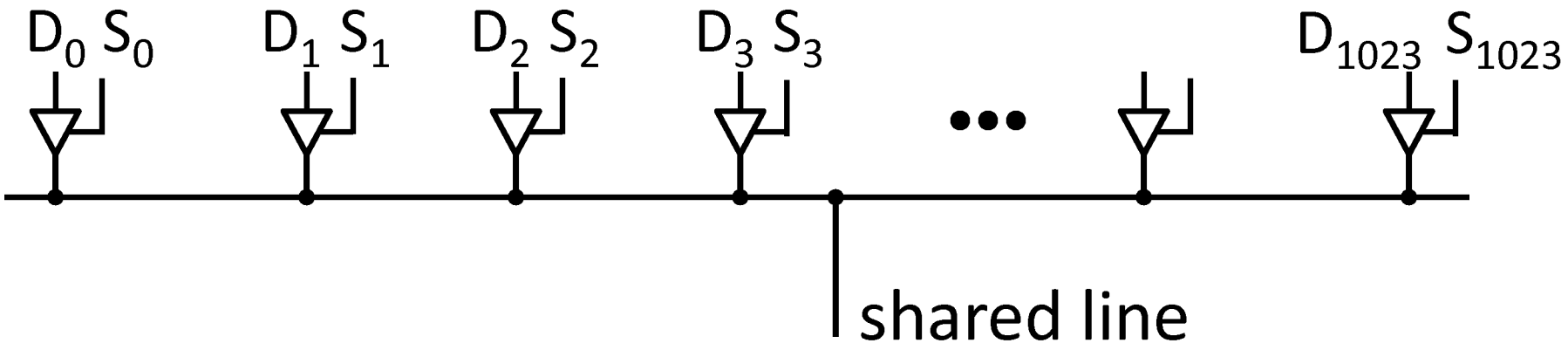
- Many FFs/outputs/etc. connected to single wire
- Only one output *drives* the bus at a time

Tri-State Buffers



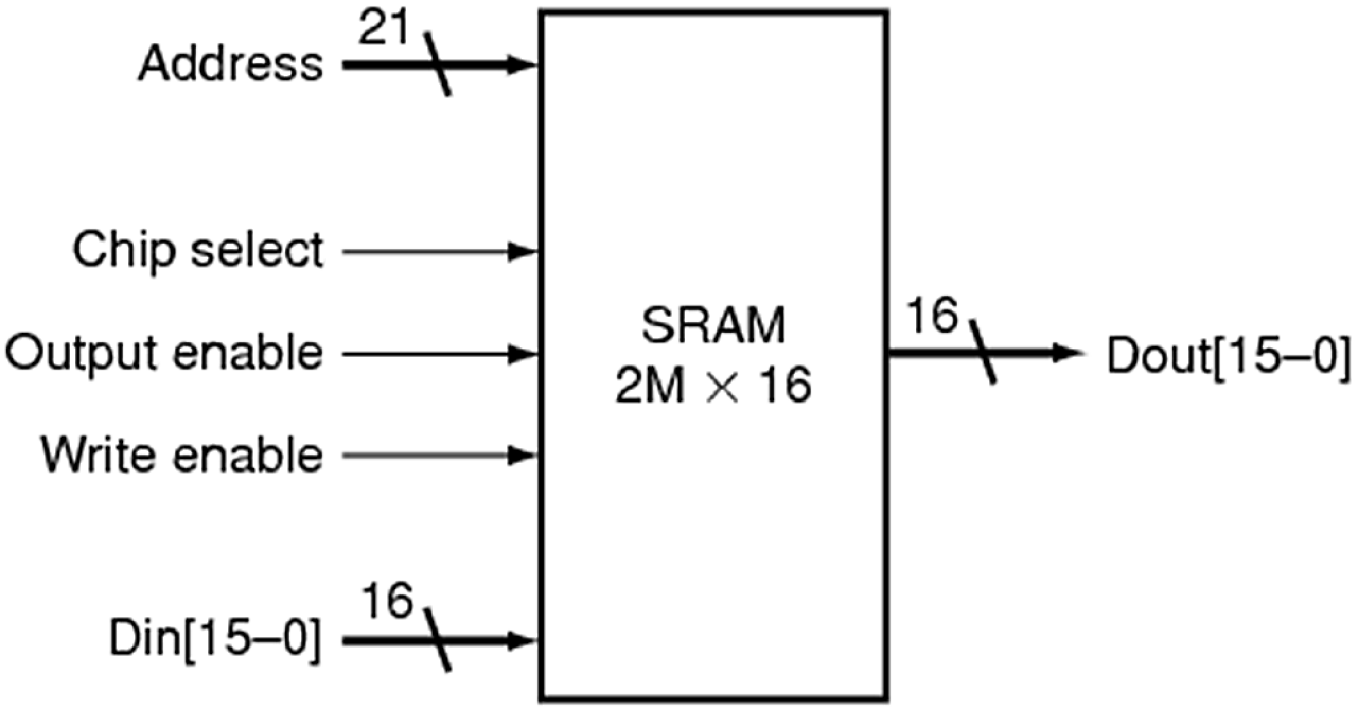
E	D	Q
0	0	z
0	1	z
1	0	0
1	1	1

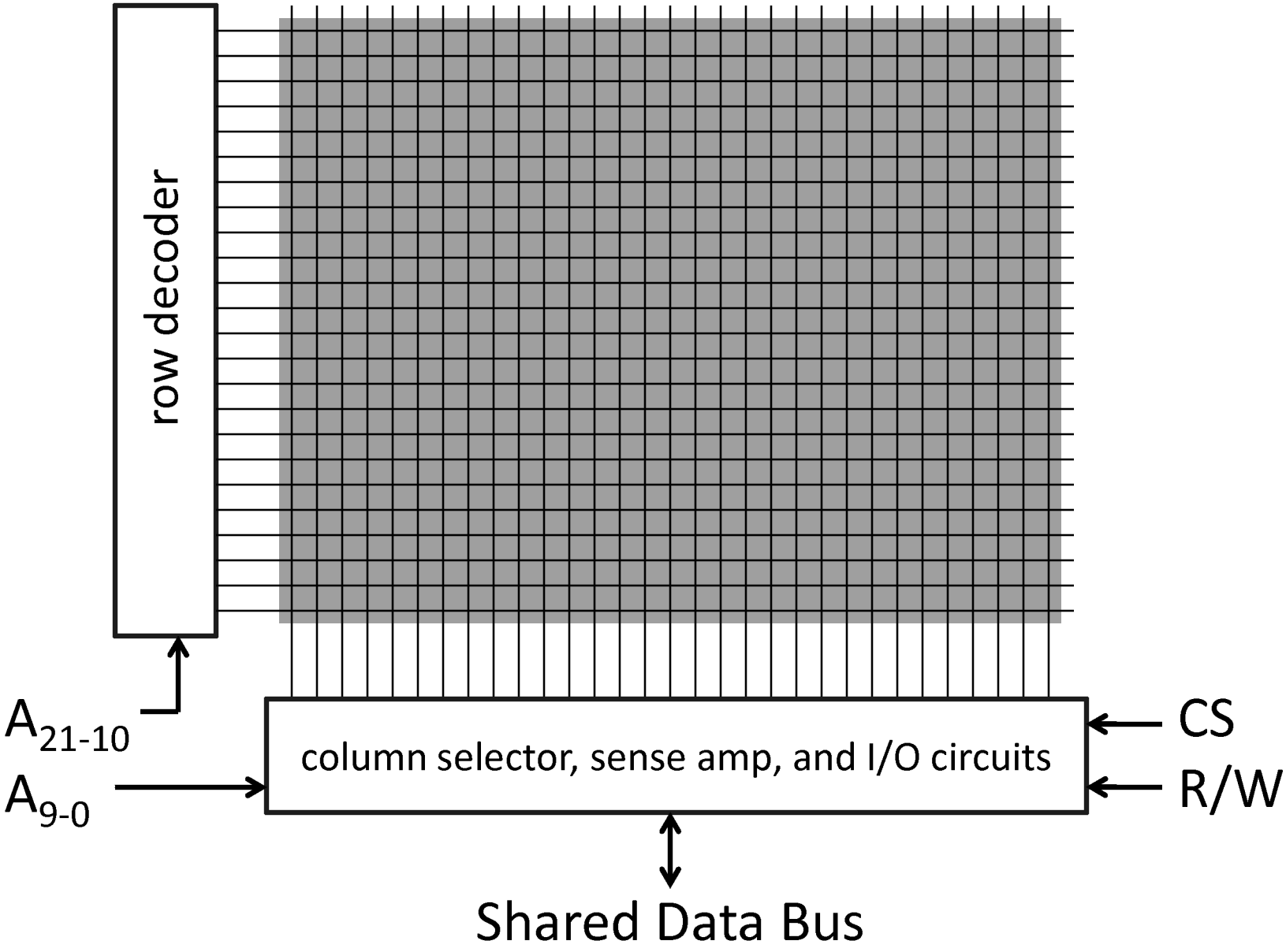



D_0 S_0
D_1 S_1
D_2 S_2
D_3 S_3
D_{1023} S_{1023}
shared line

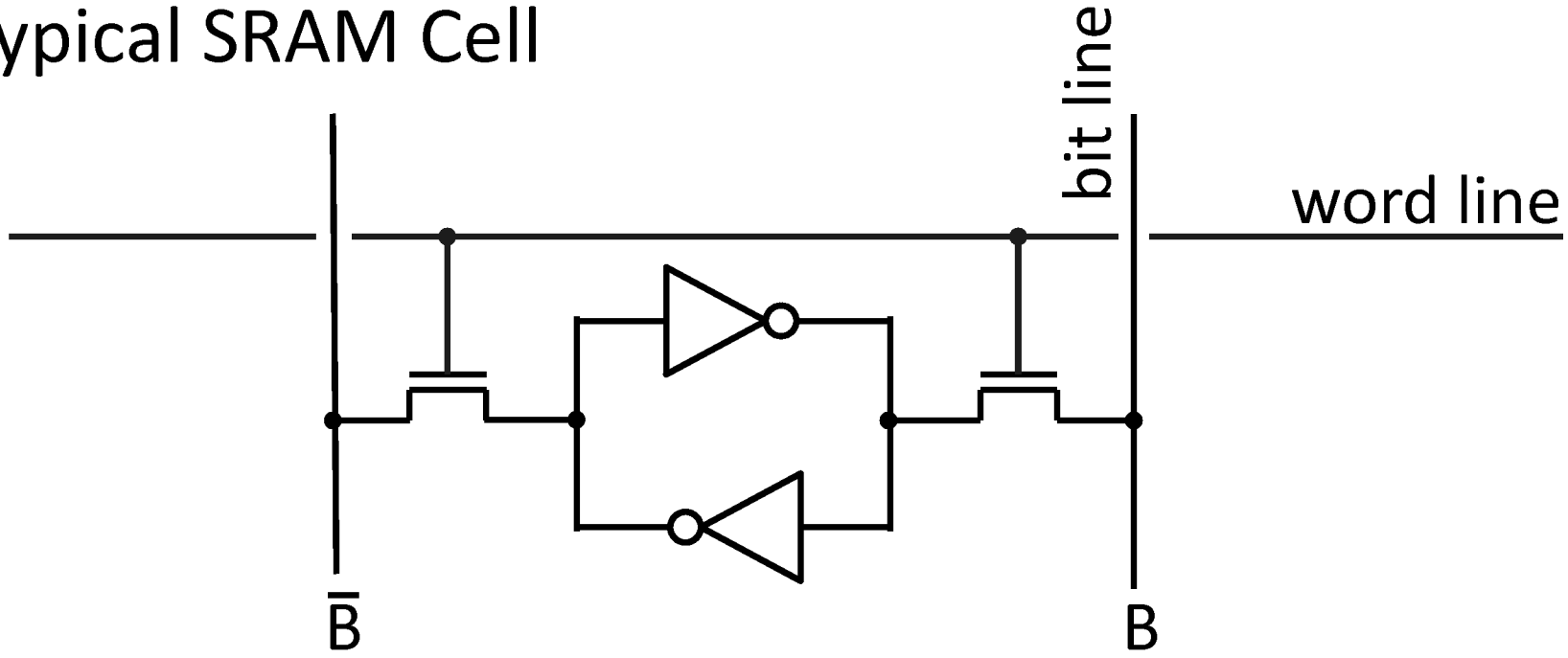
Static RAM (SRAM)

- Essentially just SR Latches + tri-states buffers




row decoder
A_{21-10}
A_{9-0}
column selector, sense amp, and I/O circuits
CS
R/W
Shared Data Bus

Typical SRAM Cell



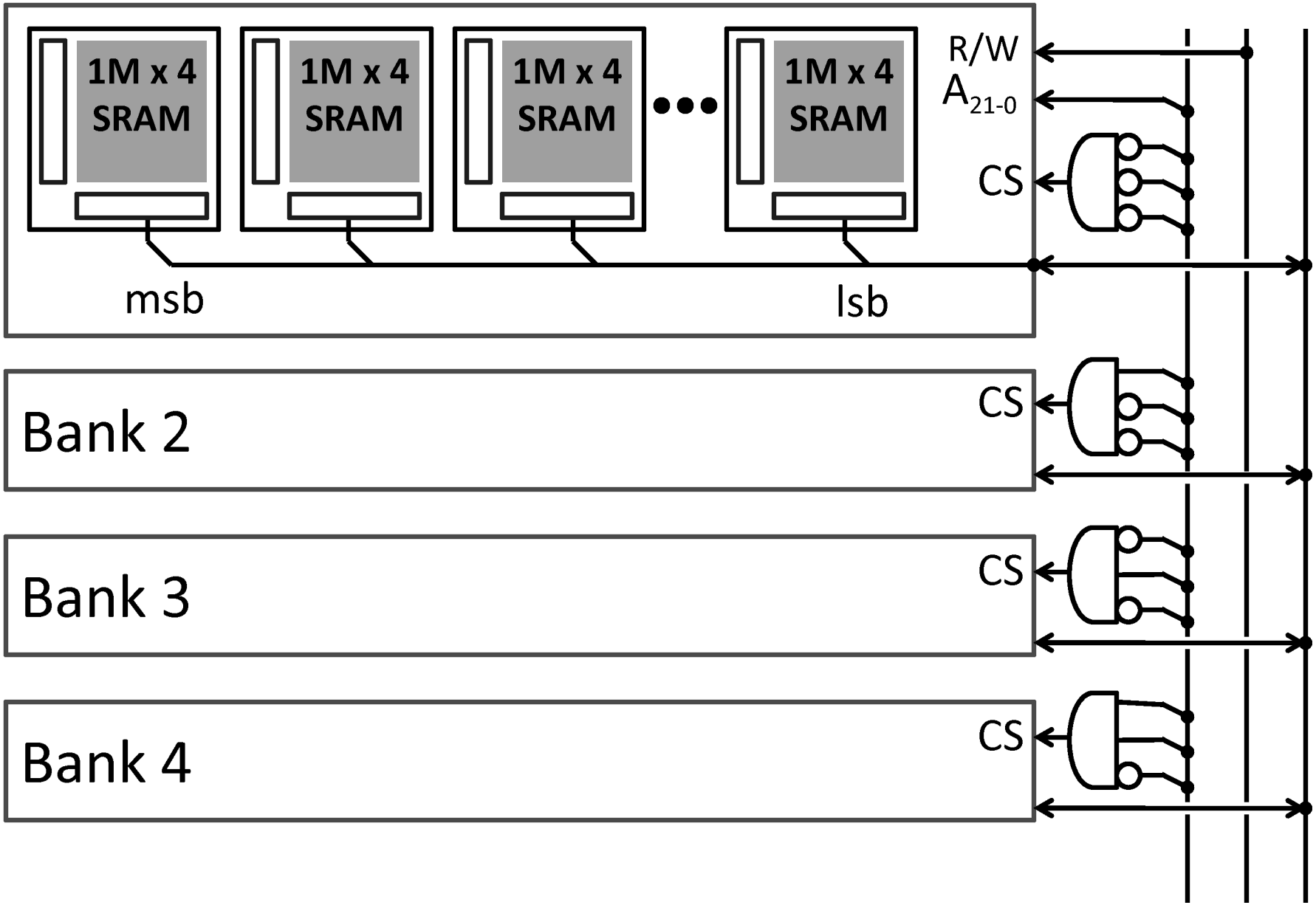
Each cell stores one bit, and requires 4 – 8 transistors (6 is typical)

Read:

- pre-charge B and $\overline{B}$ to Vdd/2
- pull word line high
- cell pulls B or $\overline{B}$ low, sense amp detects voltage difference

Write:

- pull word line high
- drive B and $\overline{B}$ to flip cell

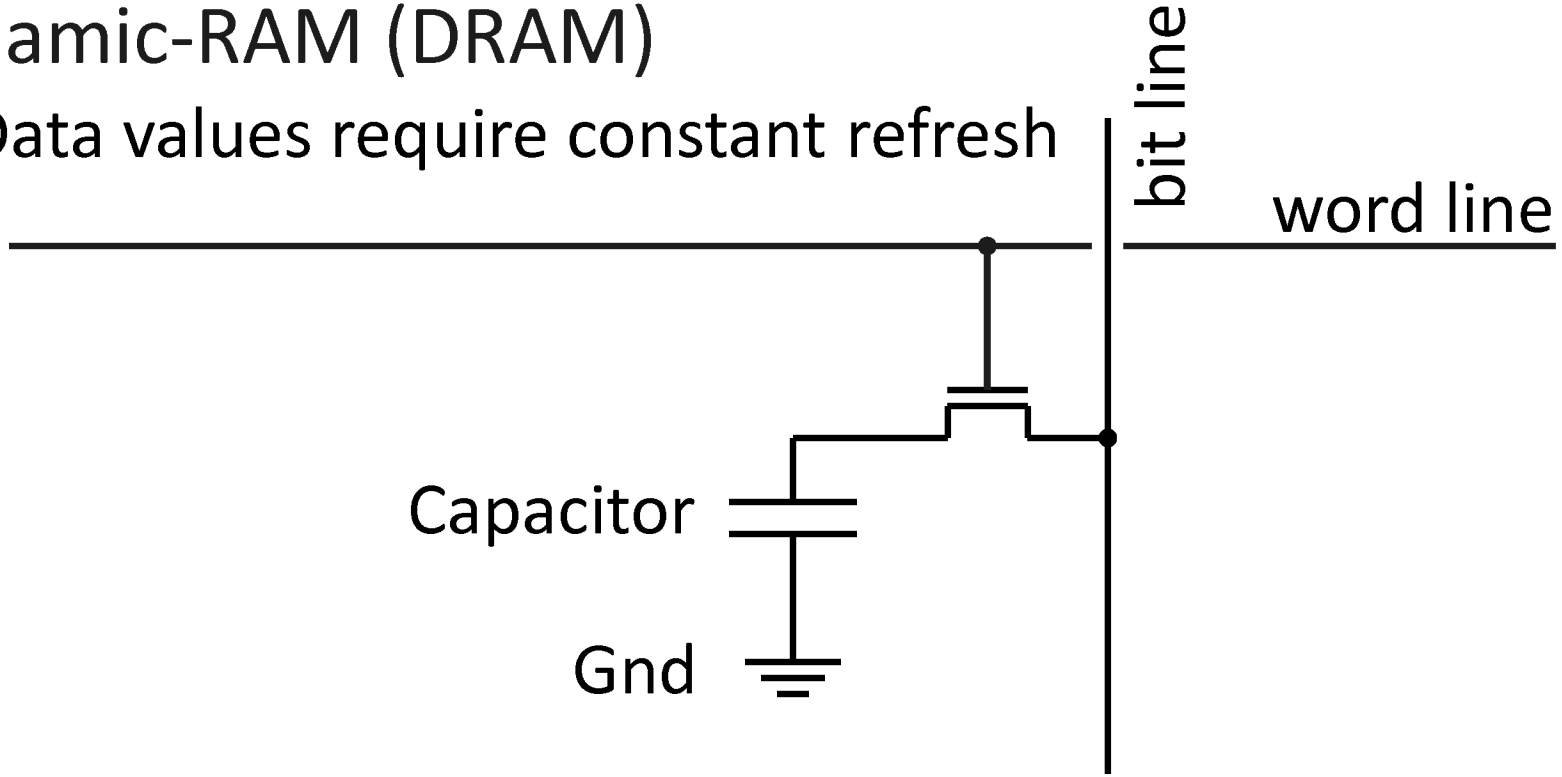

1M x 4 SRAM
1M x 4 SRAM
1M x 4 SRAM
1M x 4 SRAM
R/W
A21-0
CS
msb
lsb
Bank 2
CS
Bank 3
CS
Bank 4
CS

SRAM

- A few transistors (~6) per cell
- Used for working memory (caches)
- But for even higher density…

Dynamic-RAM (DRAM)

- Data values require constant refresh



Single transistor vs. many gates

- Denser, cheaper ($30/1GB vs. $30/2MB)
- But more complicated, and has analog sensing

Also needs refresh

- Read and write back…
- …every few milliseconds
- Organized in 2D grid, so can do rows at a time
- Chip can do refresh internally

Hence… slower and energy inefficient

Register File tradeoffs

- \+ Very fast (a few gate delays for both read and write)
- \+ Adding extra ports is straightforward
- – Expensive, doesn't scale
- – Volatile

Volatile Memory alternatives: SRAM, DRAM, …

- – Slower
- \+ Cheaper, and scales well
- – Volatile

Non-Volatile Memory (NV-RAM): Flash, EEPROM, …

- \+ Scales well
- – Limited lifetime; degrades after 100000 to 1M writes

We now have enough building blocks to build machines that can perform non-trivial computational tasks

Register File: Tens of words of working memory

SRAM: Millions of words of working memory

DRAM: Billions of words of working memory

NVRAM: long term storage
(usb fob, solid state disks, BIOS, …)