Memory

Can be categorized into:

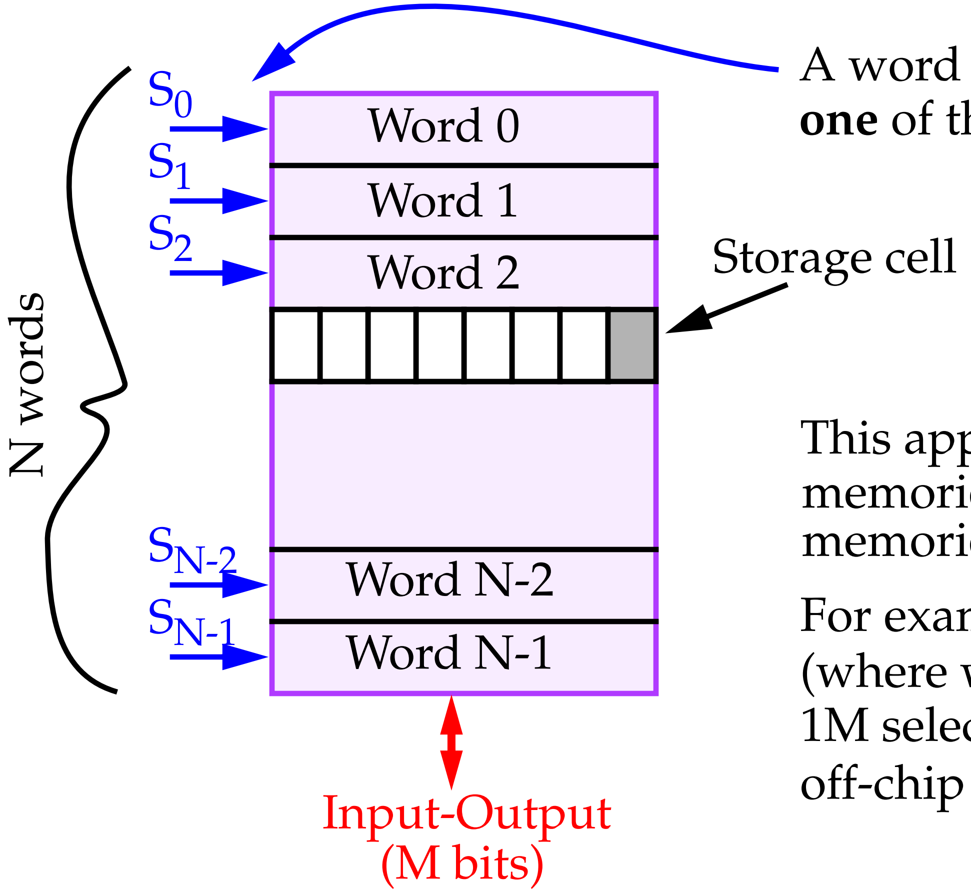
- **Read Write Memory** (RWM)
  - Random Access Memory (RAM): static SRAM (faster) verses dynamic DRAM (smaller) structures possible. Access time independent of physical location of data.
  - Non-RAM: Serial Access Memory (FIFO, LIFO, Shift register) and Content Access Memory (CAM). Non-uniform access time.
- **Non-volatile Read Write Memory** (NVRWM): write time much larger than read time.
  - EPROM, E^2PROM, FLASH
- **Read Only Memory** (ROM)

A second classification for RAMs and ROMs:

- Static-load: no clock required.
- Synchronous: require a clock edge to enable memory operation.
- Asynchronous: recognize address changes and output new data. More difficult to build.

Memory Architecture

In order to build an *N-word* memory where each word is *M bits* wide (typically 1, 4 or 8 bits), a straightforward approach is to stack memory:



A word is selected by setting exactly **one** of the select bits, S_x, high.

This approach works well for small memories but has problems for large memories.

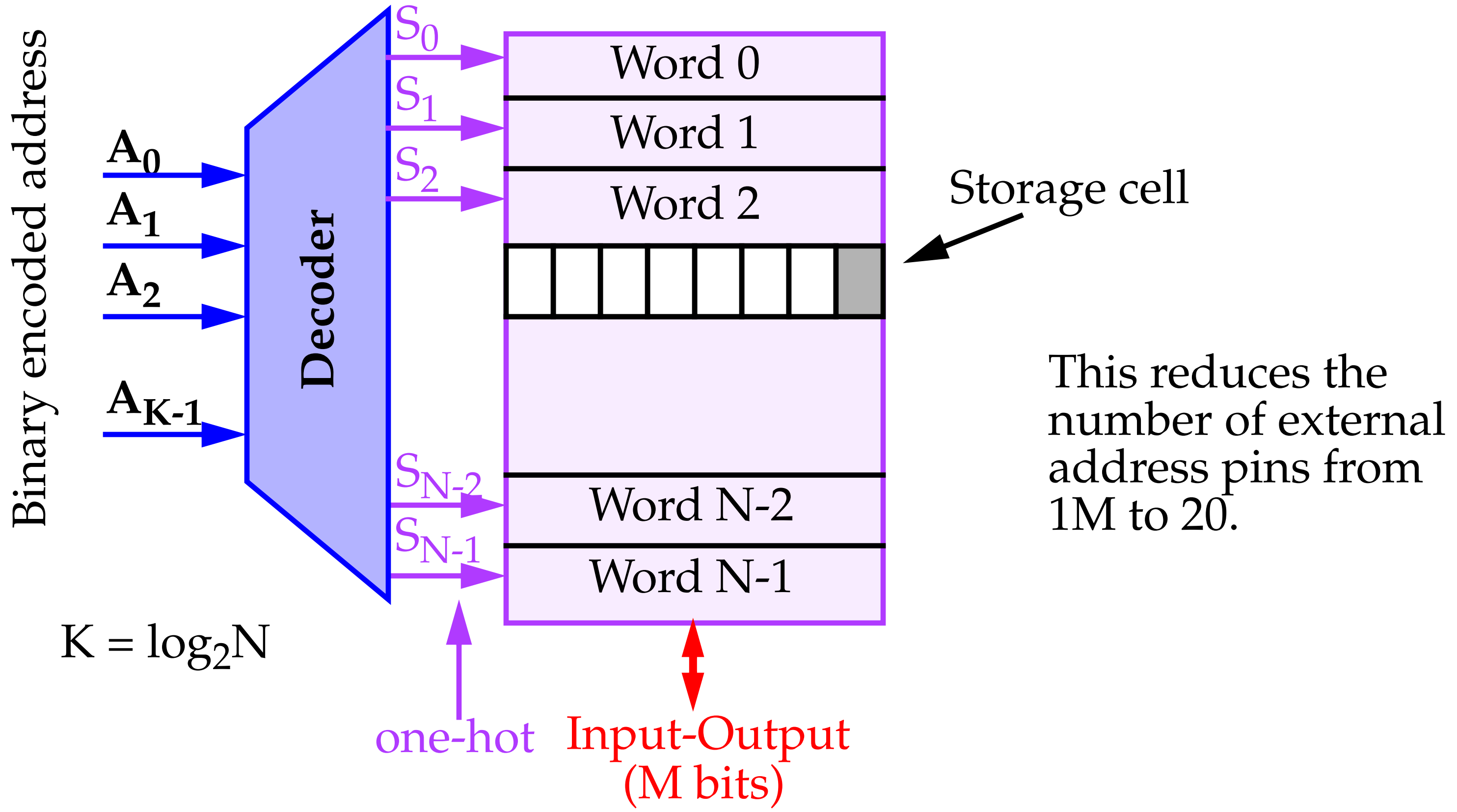
For example, to build a 1Mword (where word = 8 bits) memory, requires 1M select lines, provided by some off-chip device.

This approach is not practical.

What can we do?

Memory Architecture

Add a decoder to solve the package problem:



This reduces the number of external address pins from 1M to 20.

This does not address the **memory aspect ratio** problem:

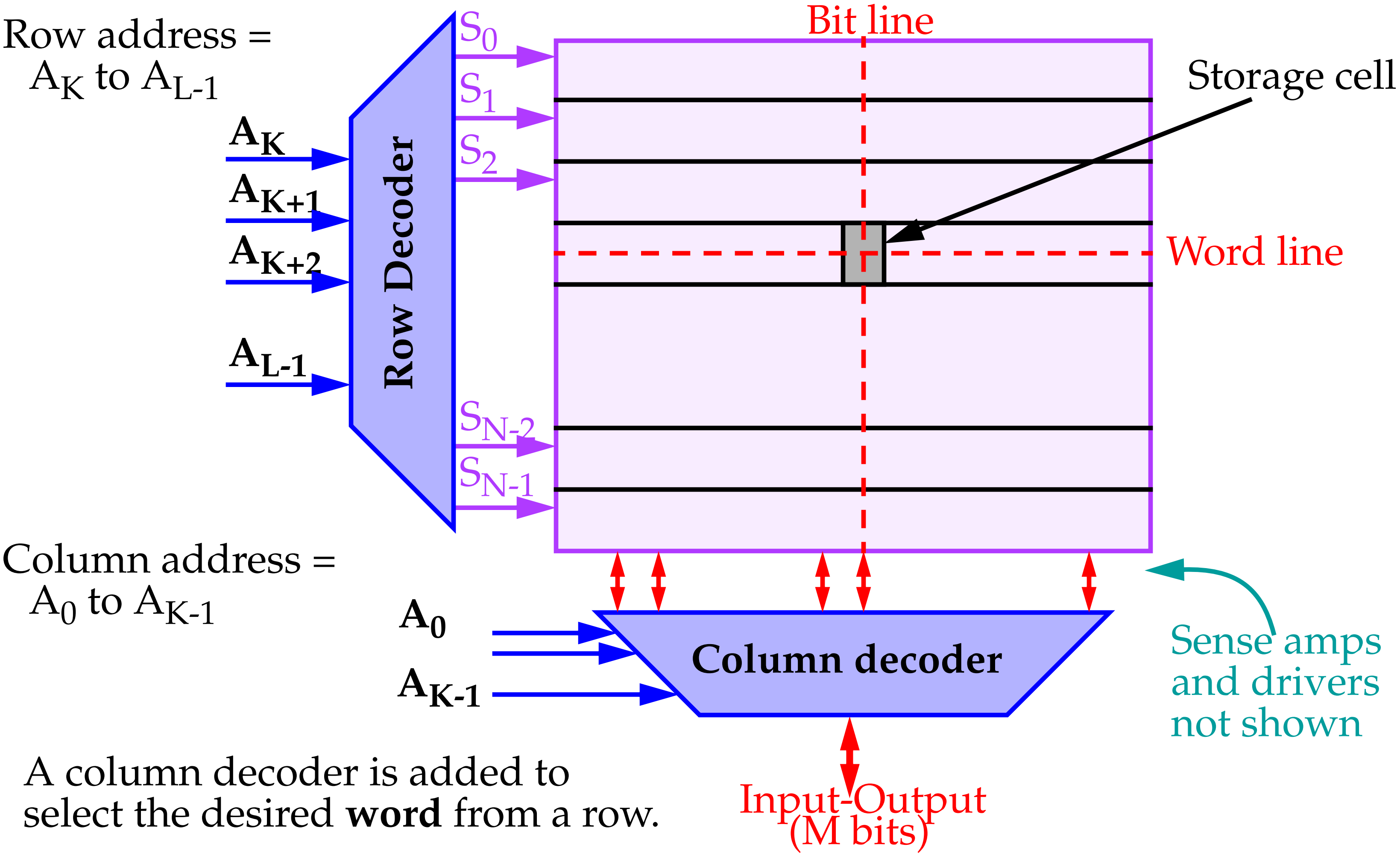
The memory is 128,000 time higher than wide ($2^{20}/2^3$) !

Besides the bizarre shape factor, the design is *extremely slow* since the vertical wires are VERY long (delay is at least linear to length).

Memory Architecture

The vertical and horizontal dimensions are usually very similar, for an aspect ratio of *unity*.

Multiple words are stored in each row and selected simultaneously:



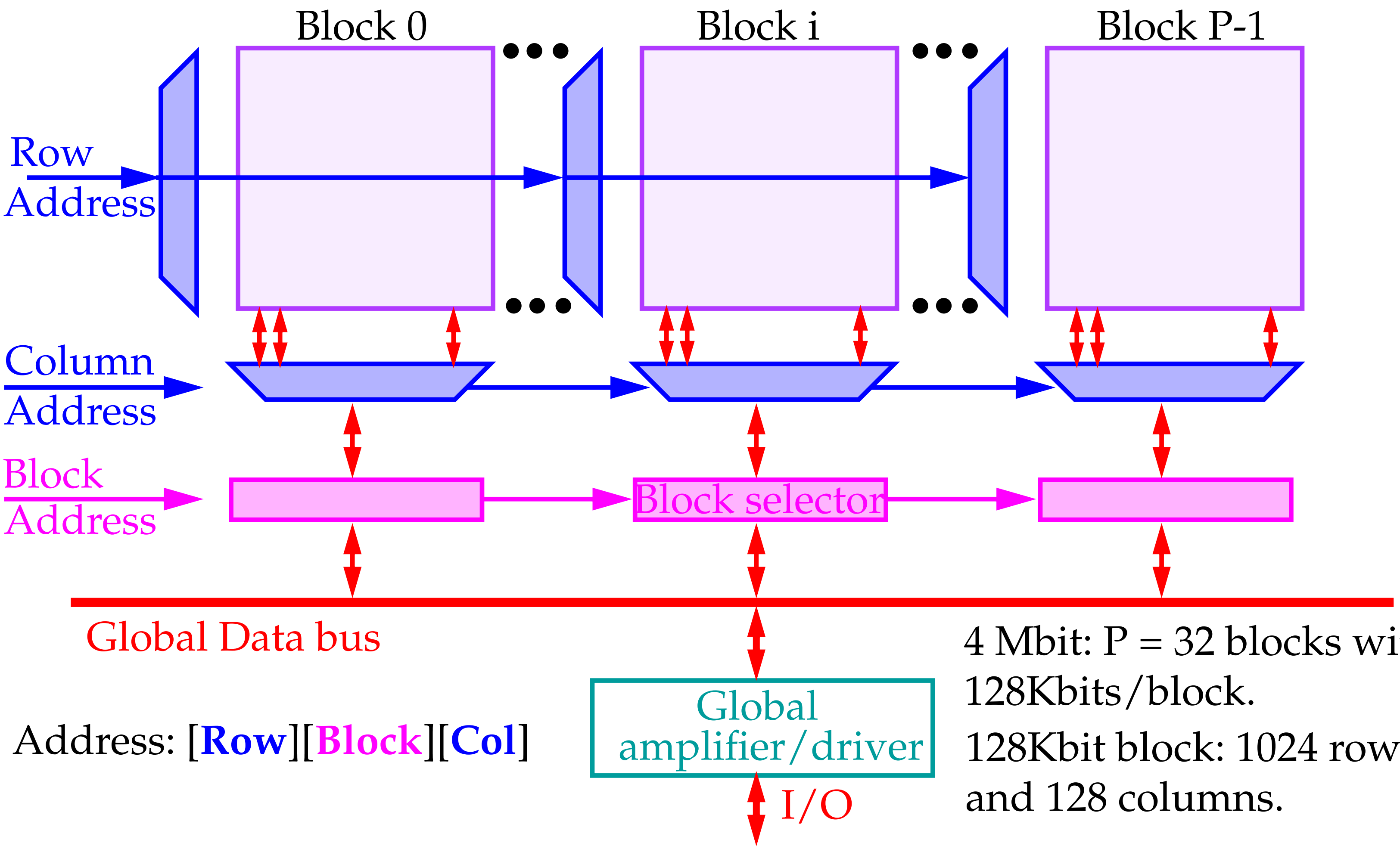
A column decoder is added to select the desired **word** from a row.

Memory Architecture

This strategy works well for memories up to 64 Kbits to 256 Kbits.

Larger memories start to suffer excess delay along bit and word lines.

A **third dimension** is added to the address space to solve this problem:



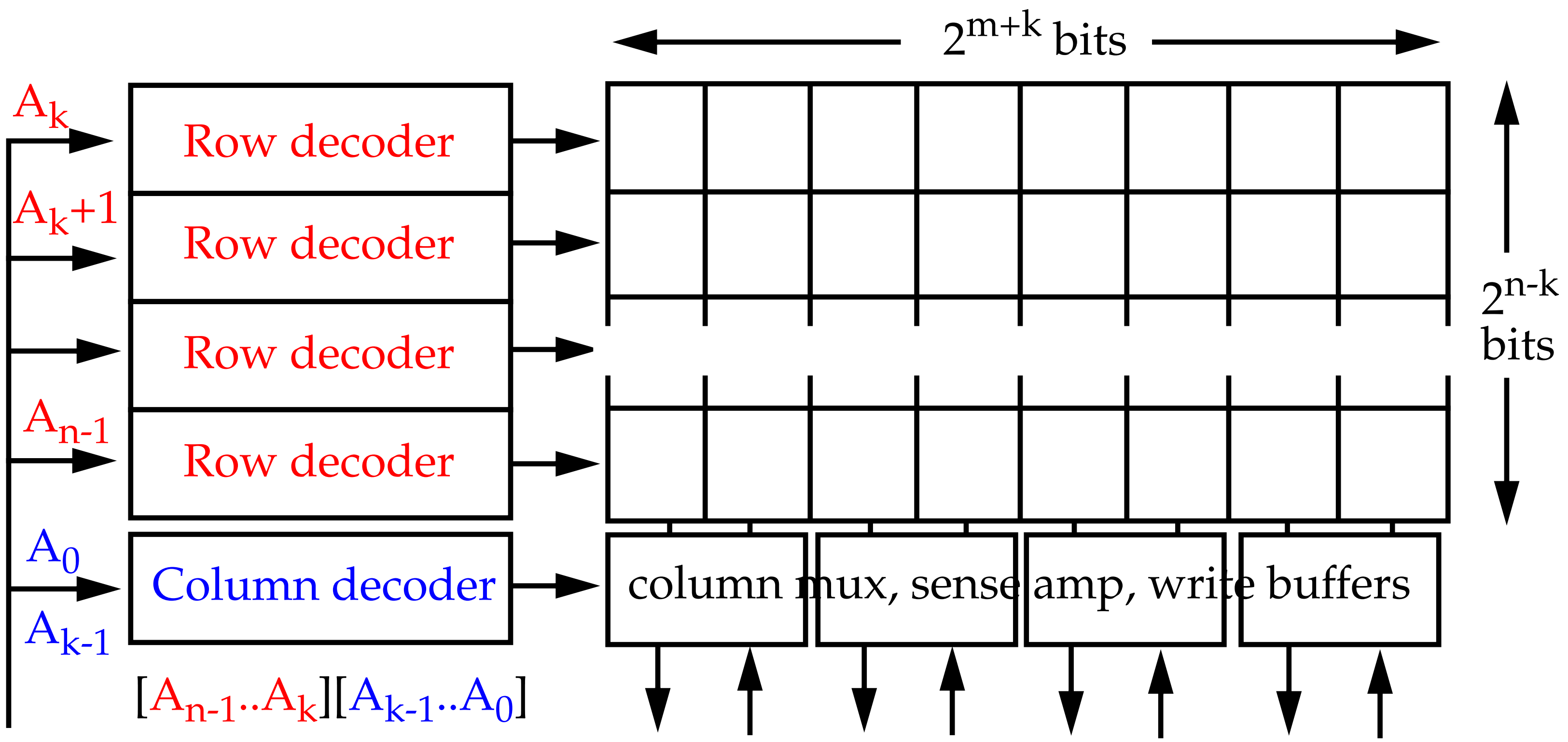
Address: [**Row**][**Block**][**Col**]

4 Mbit: P = 32 blocks with 128Kbits/block.

128Kbit block: 1024 rows and 128 columns.

Memory: Architecture

An example:

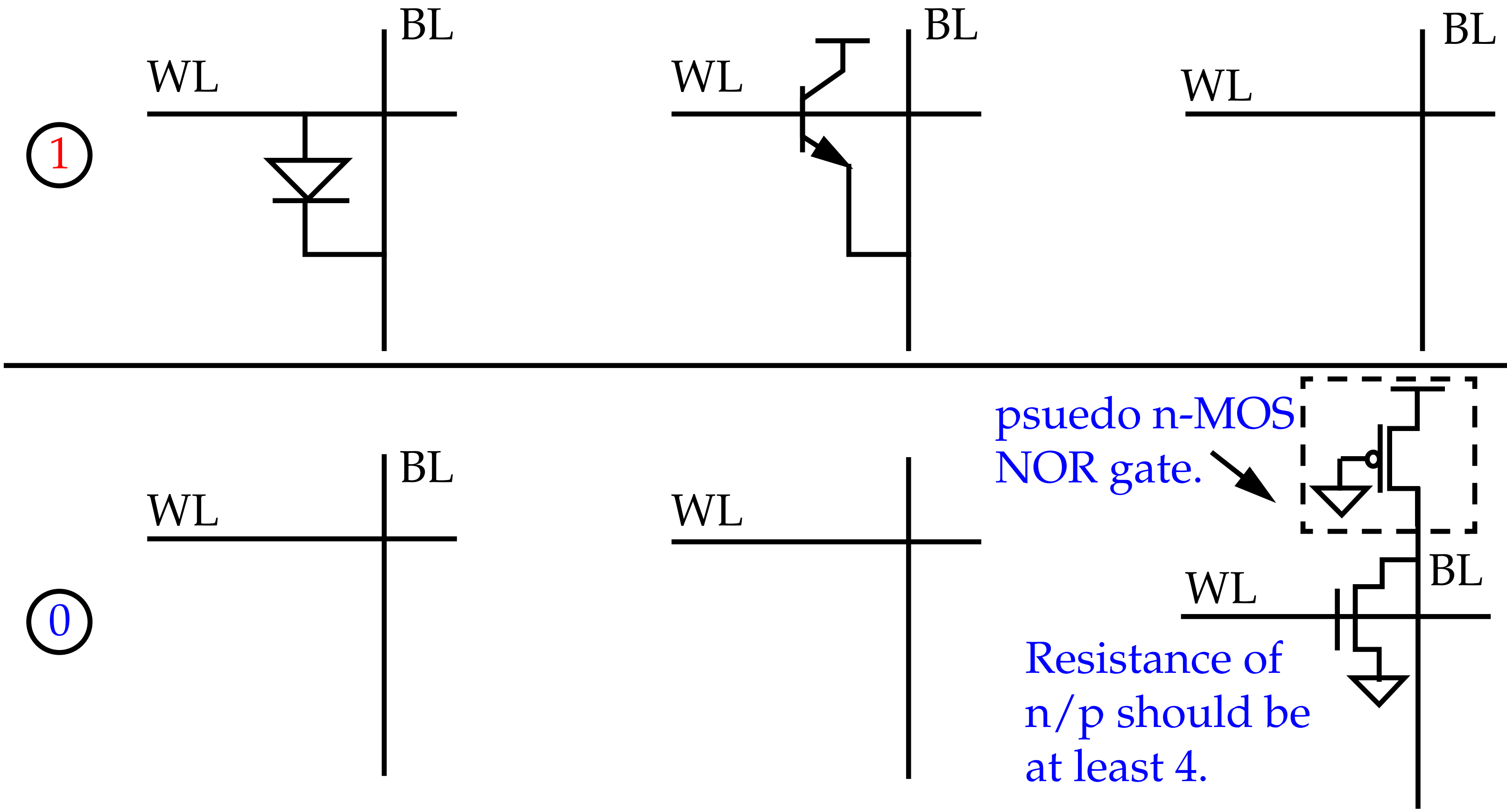


For example: Let N = 1,048,576 and M = 8 bits for a 1 million byte memory. n = $\log_2 N$ = 20, k = 8 and m = $\log_2 M$ = 3.

Then there are 2^{n-k} rows = 2^{12} = 4096 and 2^{k+m} columns/2^3 bits per word = 2^8 = 256 words.

ROM

ROM cells are permanently fixed: Several possibilities:



psuedo n-MOS NOR gate.

Resistance of n/p should be at least 4.

Diode supplies current to raise BL (bitline) for all cells on the row.

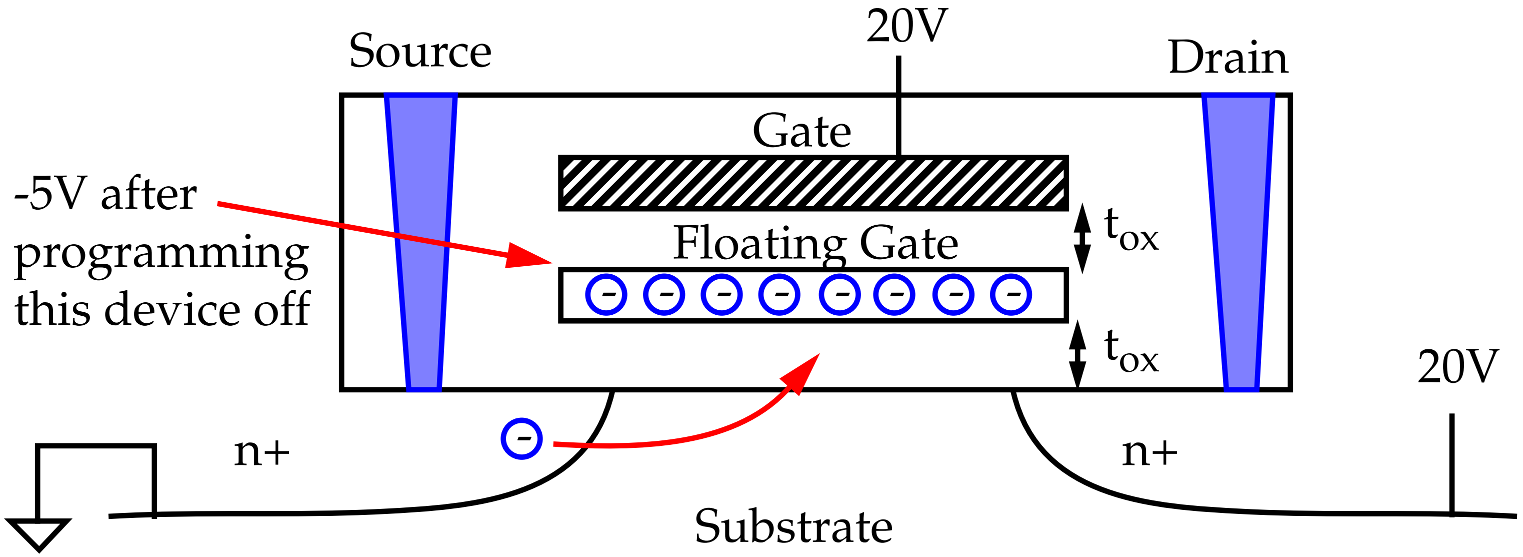
BJT supplies current to raise BL for each cell on the row. Requires V_{DD} to be routed.

p-MOS used to hold BL high. n-MOS provides pull-down path.

Non-volatile Read-Write Memories

Virtually identical in structure to ROMs.

Selective enabling/disabling of transistors is accomplished through modifications to **threshold voltage**. This is accomplished through a floating gate.



Applying a high voltage (15 to 20 V) between source and gate-drain create high electric field and causes avalanche injection to occur.

Hot electrons traverse first oxide and get trapped on floating gate, leaving it negatively charged.

This increases the threshold voltage to ~7V. Applying 5V to the gate does not permit the device to turn on.

Non-volatile Read-Write Memories

The method of erasing is the main differentiating factor between the various classes of reprogrammable nonvolatile memories.

- **EPROM**:

  UV light renders oxide slightly conductive.

  Erase is slow (seconds to several minutes).

  Programming is slow (5-10 microsecs per word).

  Limited number of programming cycles - about 1000.

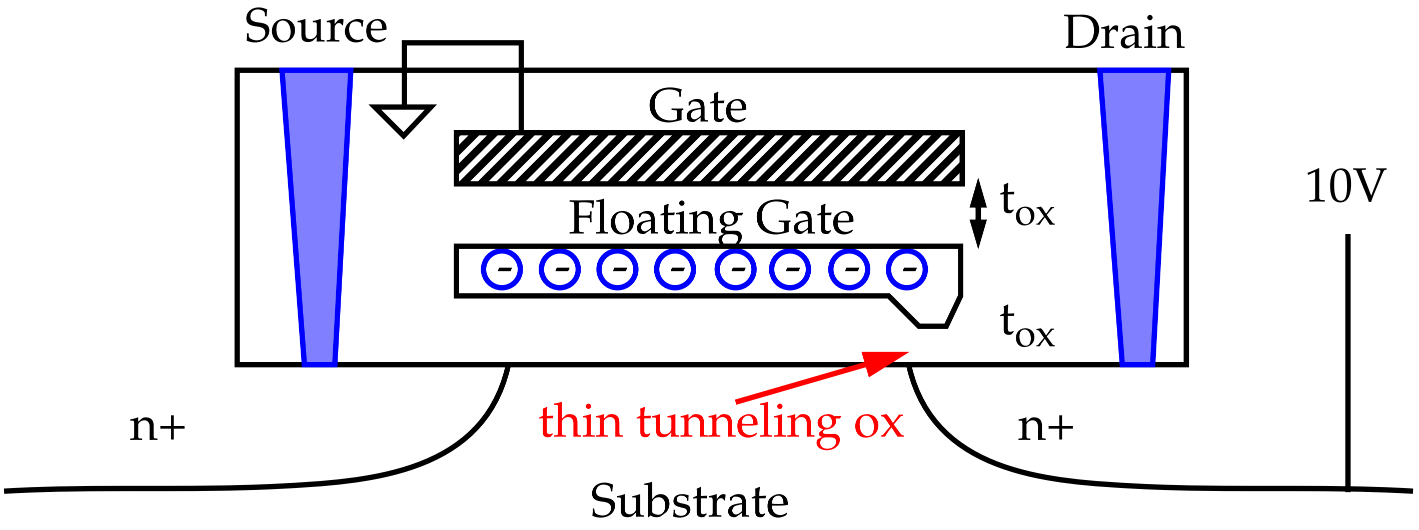
  Very dense - single transistor functions as both the programming and access device.

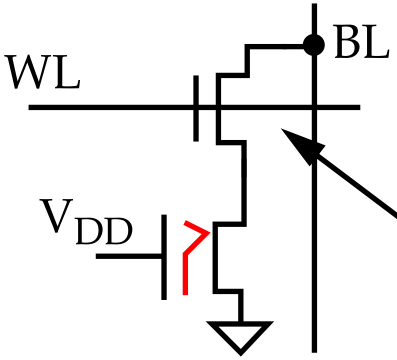
Non-volatile Read-Write Memories

- **EEPROM** or **E^{2}PROM**:

  Very thin oxide allows electrons to flow to and from the gate via Fowler-Nordheim tunneling with V_{GD} applied.

  Erasure is achieved by reversing the voltage applied during writing.





Threshold control becomes a problem:

Removing too much charge results in a depletion device that cannot be turned off.

Remedy: Add an access transistor.

Non-volatile Read-Write Memories

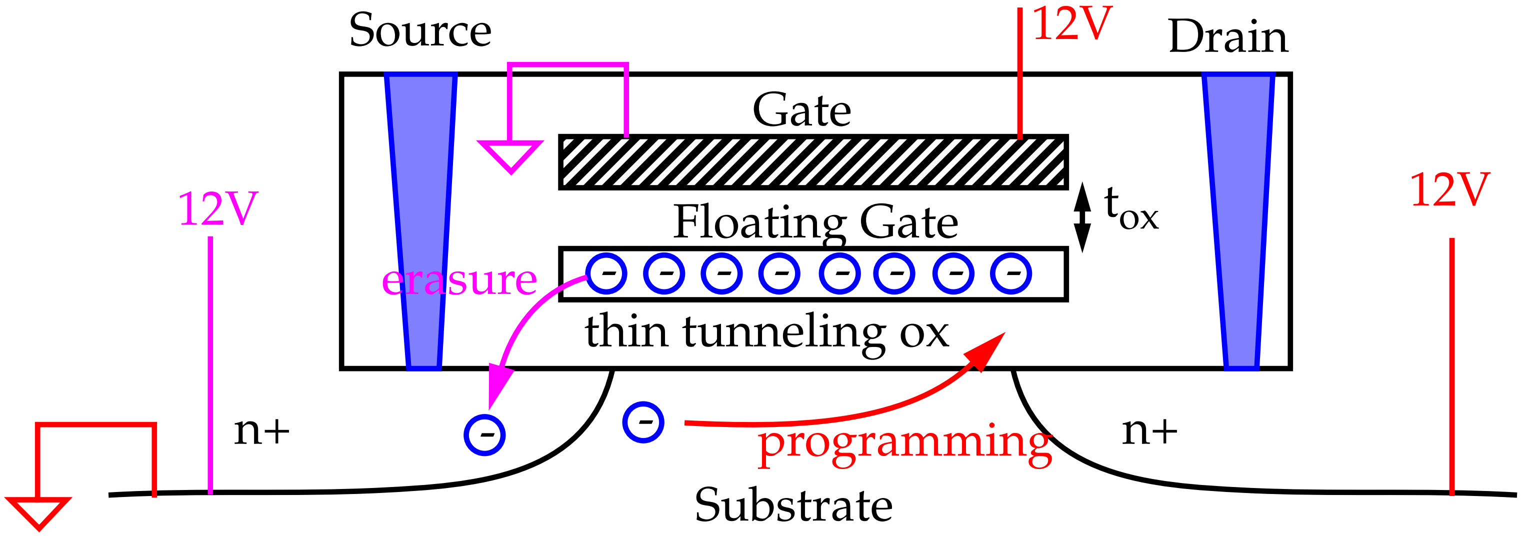
- **Flash EEPROM**:

  Combines density adv. of EPROM with versatility of EEPROM.

  Uses avalanche hot-electron-injection approach to program.

  Erasure performed using Fowler-Nordheim tunneling.

  Monitoring control hardware checks the value of the threshold during erasure - making sure the unprogrammed transistor remains an enhancement device.

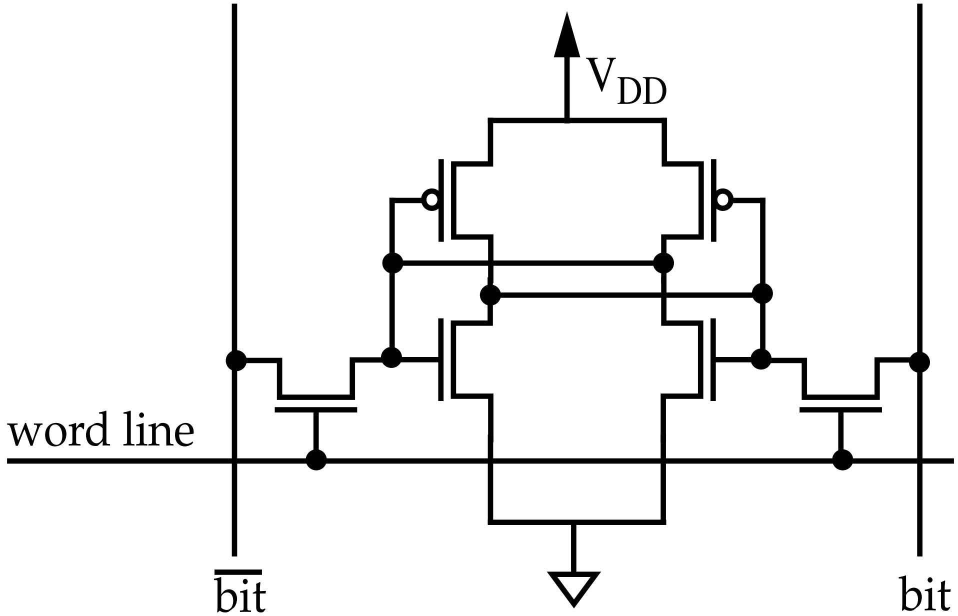


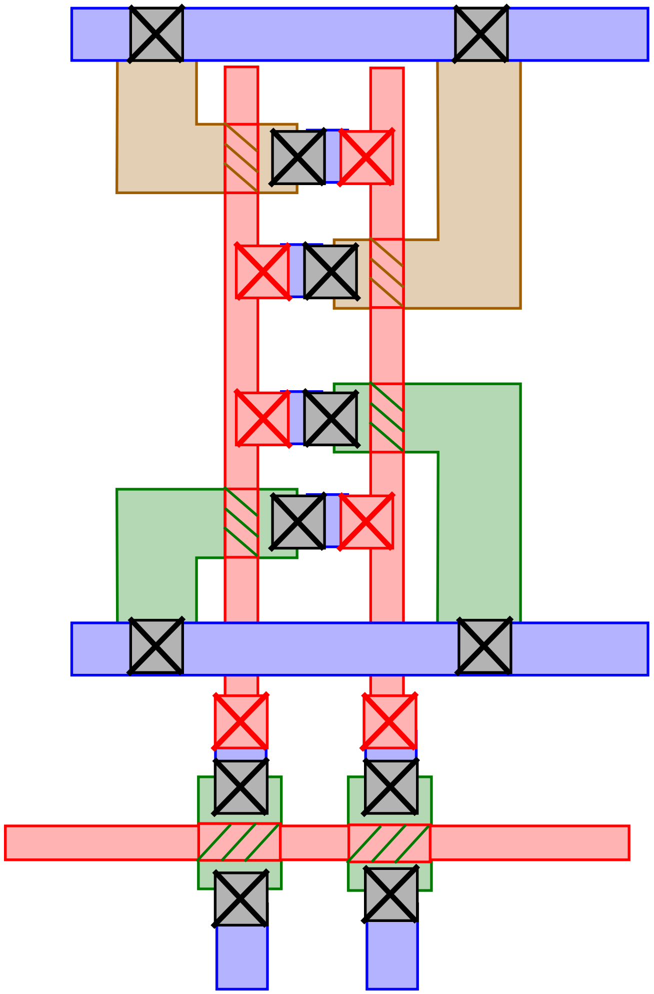
Programming performed by applying 12V to gate and drain.

Erasure performed with gate grounded and source at 12V.

Read-Write Memories (RAM)

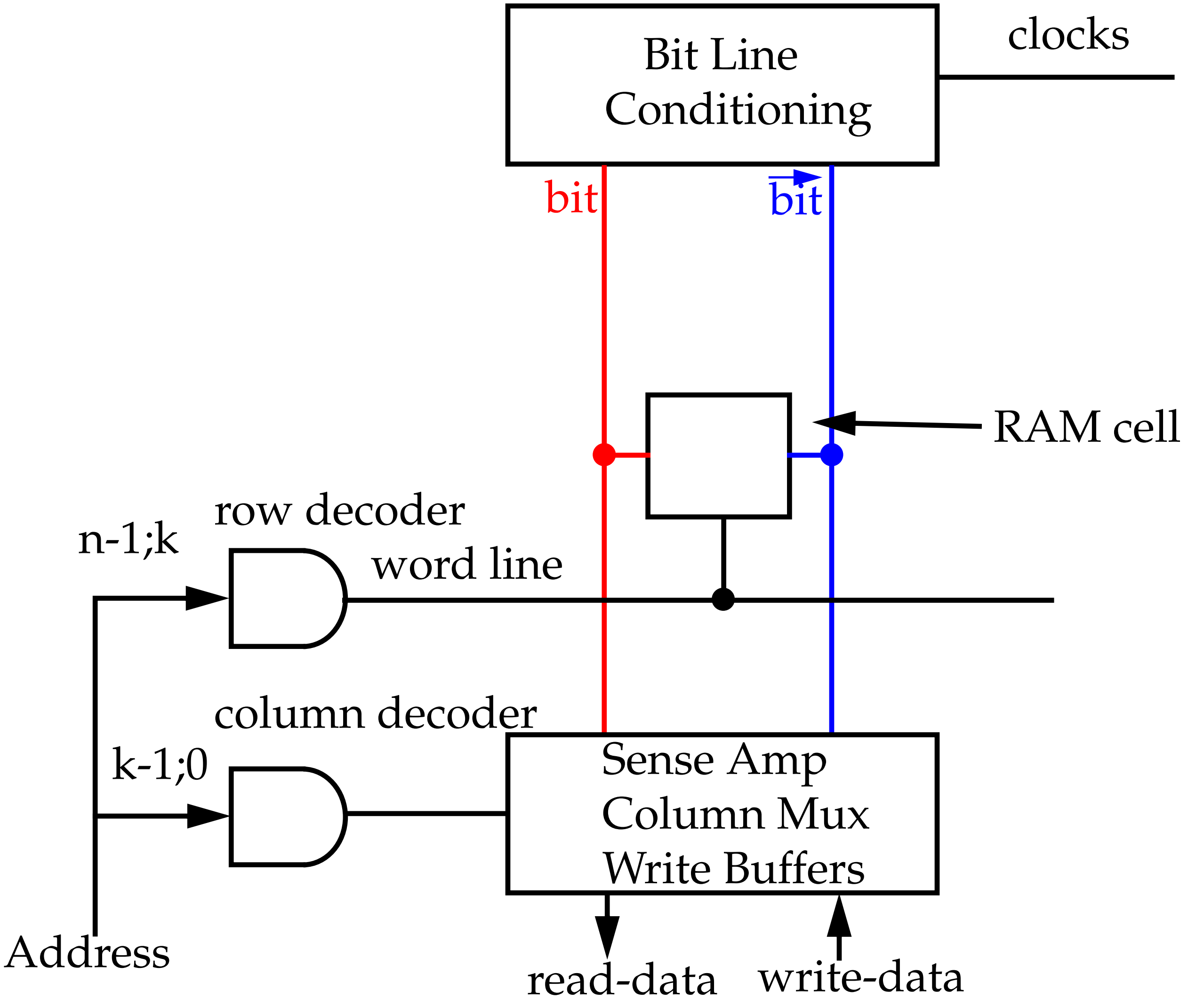
SRAM:





Read-Write Memories (RAM)

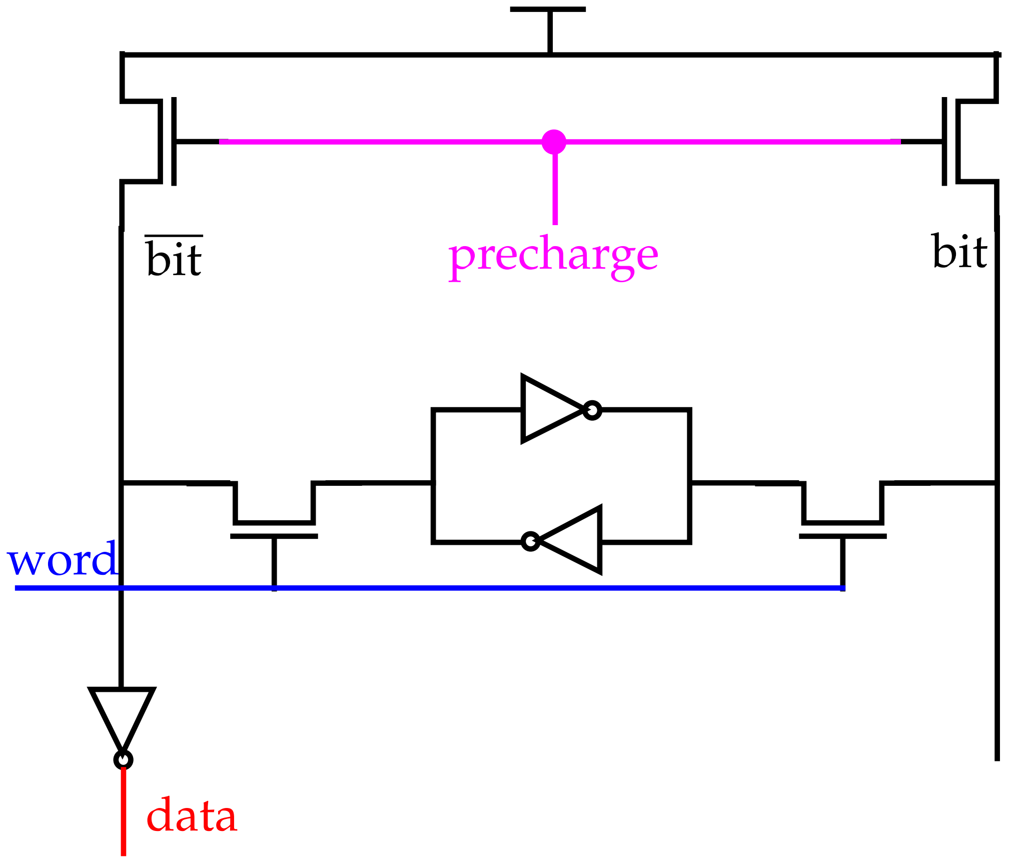
Generic RAM circuit:



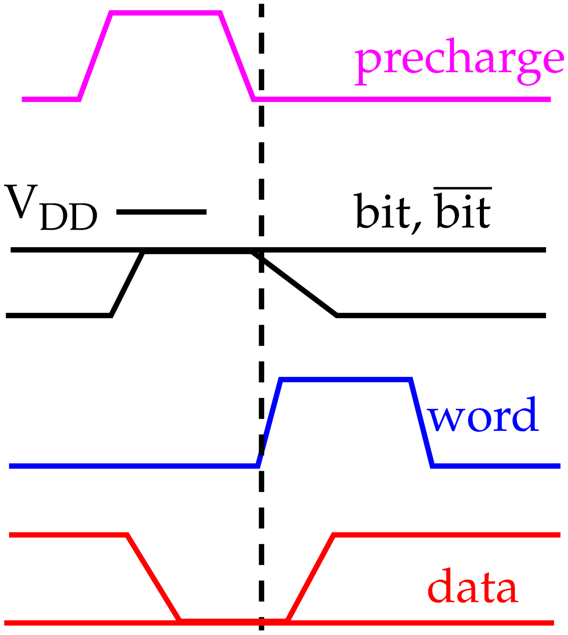
Read-Write Memories (RAM)

SRAM: Read Operation

Precharging bit and bit_bar to 5V before enabling the word line improves performance.

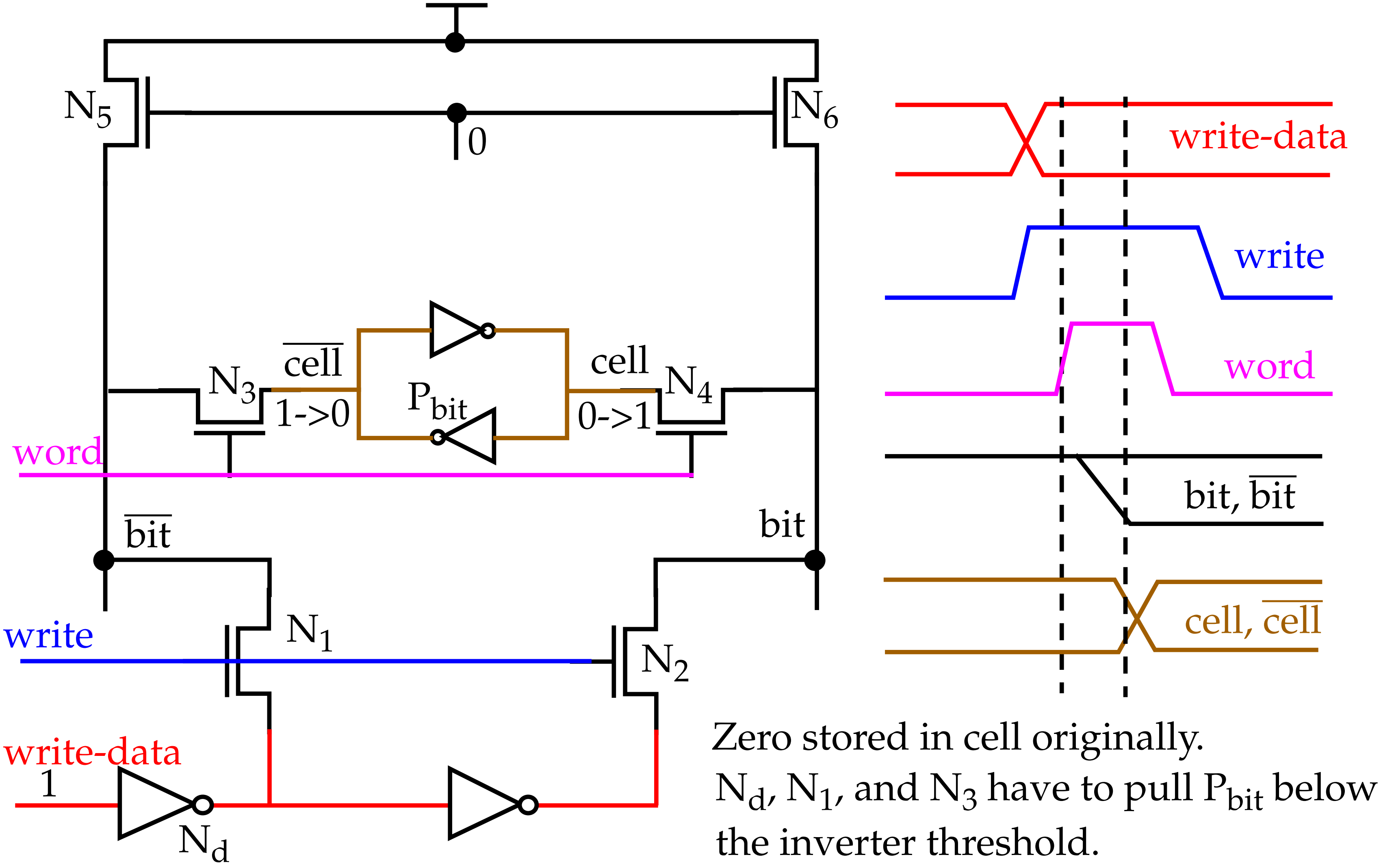


To optimize speed, use n-channels as precharge devices.



Read-Write Memories (RAM)

SRAM: Write Operation:

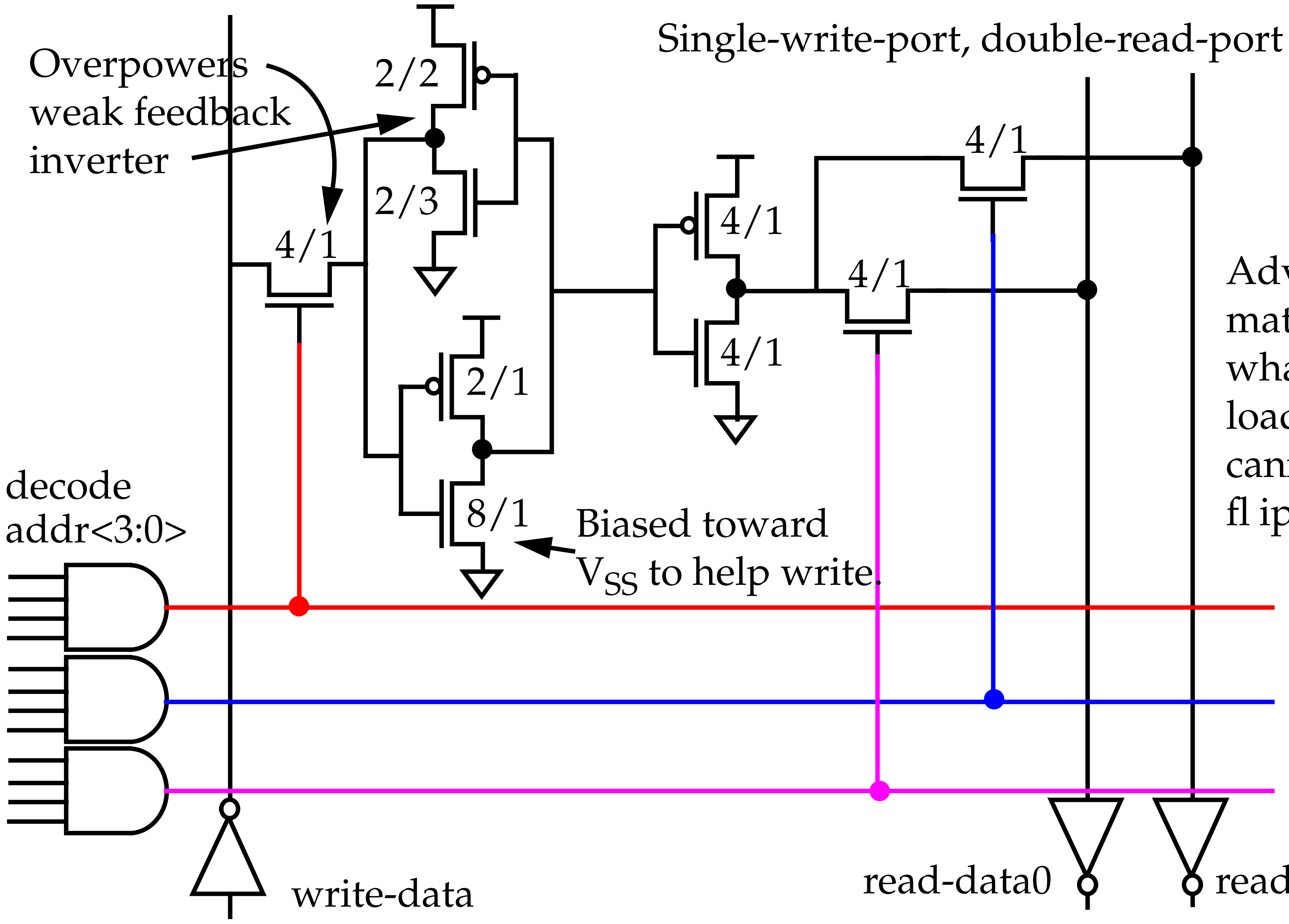


Zero stored in cell originally.

N_d, N_1, and N_3 have to pull P_{bit} below the inverter threshold.

Read-Write Memories (RAM)

Register files:


Overpowers weak feedback inverter
2/2
2/3
4/1
2/1
8/1
Biased toward V_{SS} to help write.
4/1
4/1
4/1
4/1
Single-write-port, double-read-port
decode addr<3:0>
write-data
read-data0
read-data1

Adv: No matter what the load, cell cannot be flipped.

Read-Write Memories (RAM)

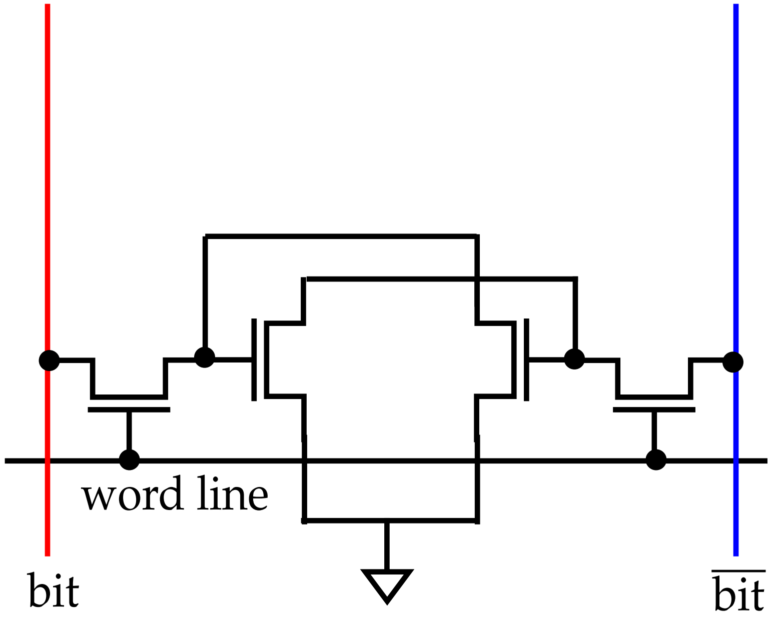
DRAM:

Refresh: Compensate for charge loss by periodically rewriting the cell contents.

Read followed by a write operation.

Typical refresh cycles occur every 1 to 4 milliseconds.

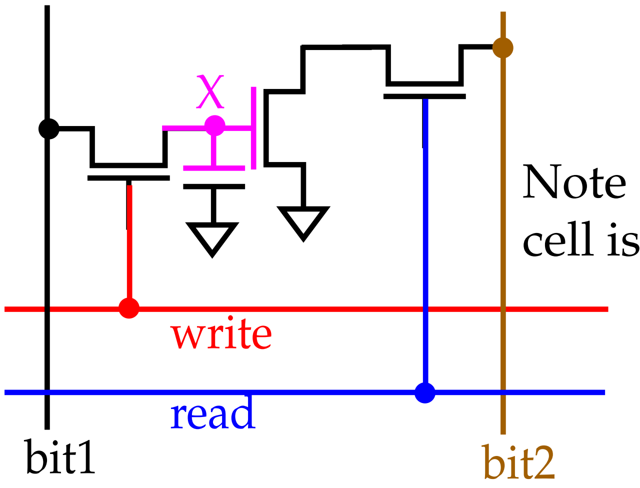
4 transistor DRAM created by simply eliminating the p tree in an SRAM cell.



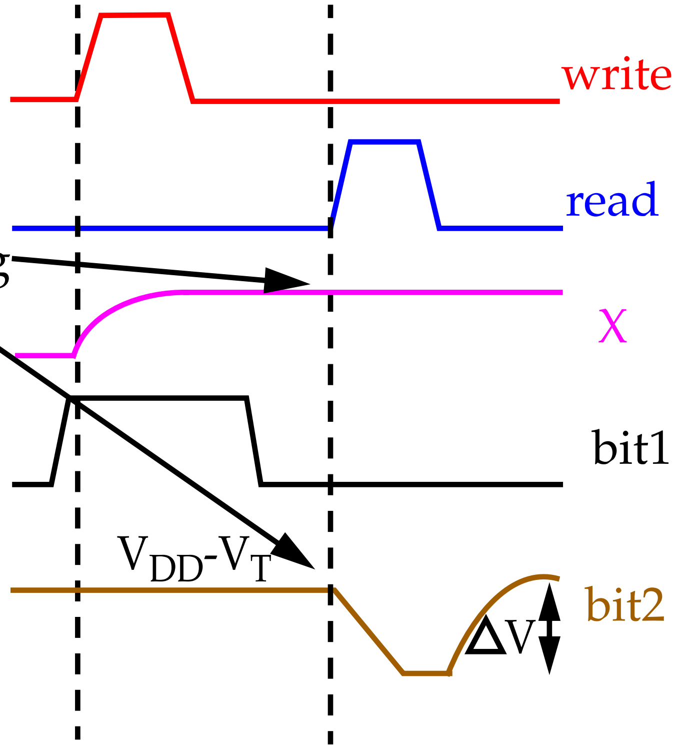
Logic 1 values are, of course, a threshold below V_{DD}.

Read-Write Memories (RAM)

3T DRAM:



Note that this cell is inverting



bit2 is either clamped to V_{DD} or is precharged to either V_{DD} or V_{DD}-V_T.

No device ratioing necessary here !

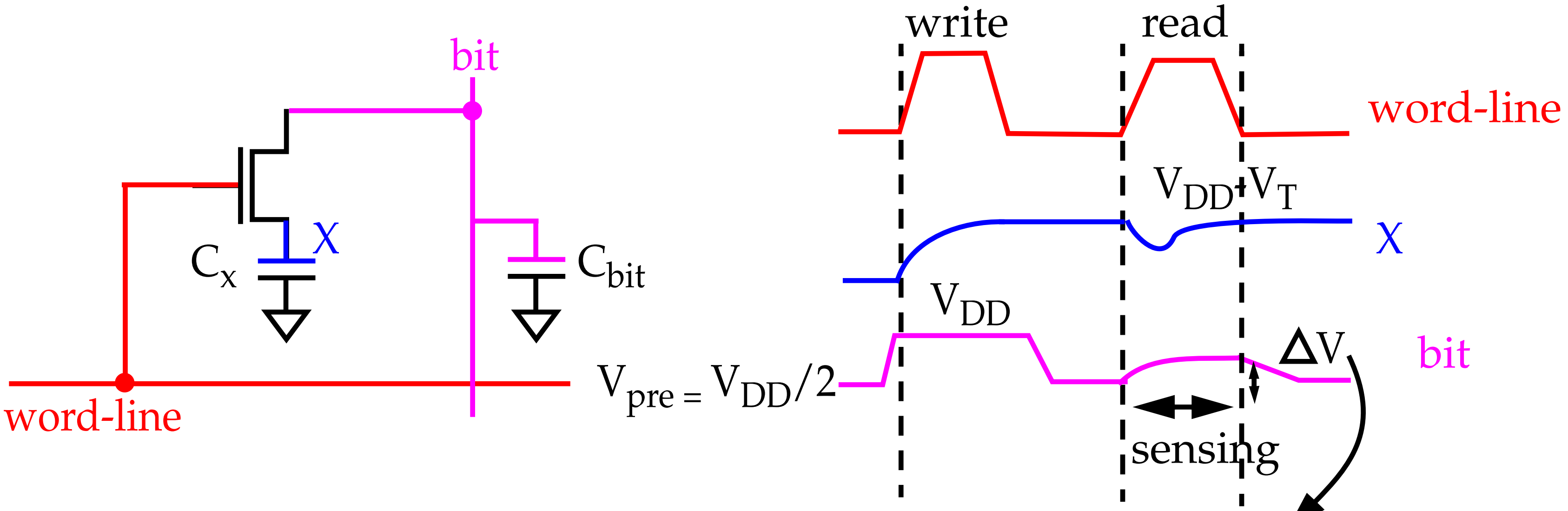
Most common method of refresh is to read *bit2*, place its inverse on *bit1* and assert *write*.

Precharge method of 'setting' *bit2* is preferred (no steady-state current).

Memory structure of choice in ASICs because of its relative simplicity in both design and operation.

Read-Write Memories (RAM)

1T DRAM



$$\Delta V = V_{bit} - V_{pre} = (V_x - V_{pre})\frac{C_x}{(C_x + C_{bit})}$$

During read operation, charge redistribution occurs between node X and node bit.

C_x is typically 1 or 2 orders of magnitude smaller than C_{bit} so the delta-V value is typically 250 mV.

Most pervasive DRAM cell in commercial memory design.

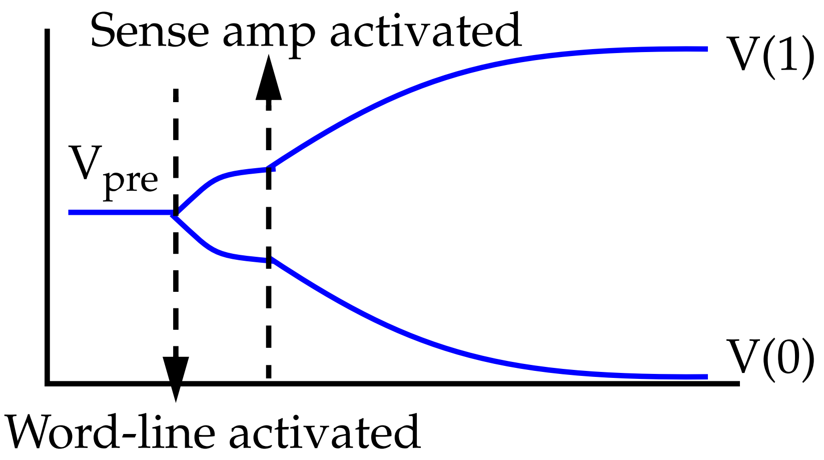
Read-Write Memories (RAM)

1T DRAM observations:

- Amplification of delta-V (through a sense amplifier) is necessary in order for the cell to be functional.

  Other cell designs used sense amps only to speed up the read operation.

- The read-out operation is destructive ! Output of sense amp is imposed onto the bit line with word-line high during read-out.



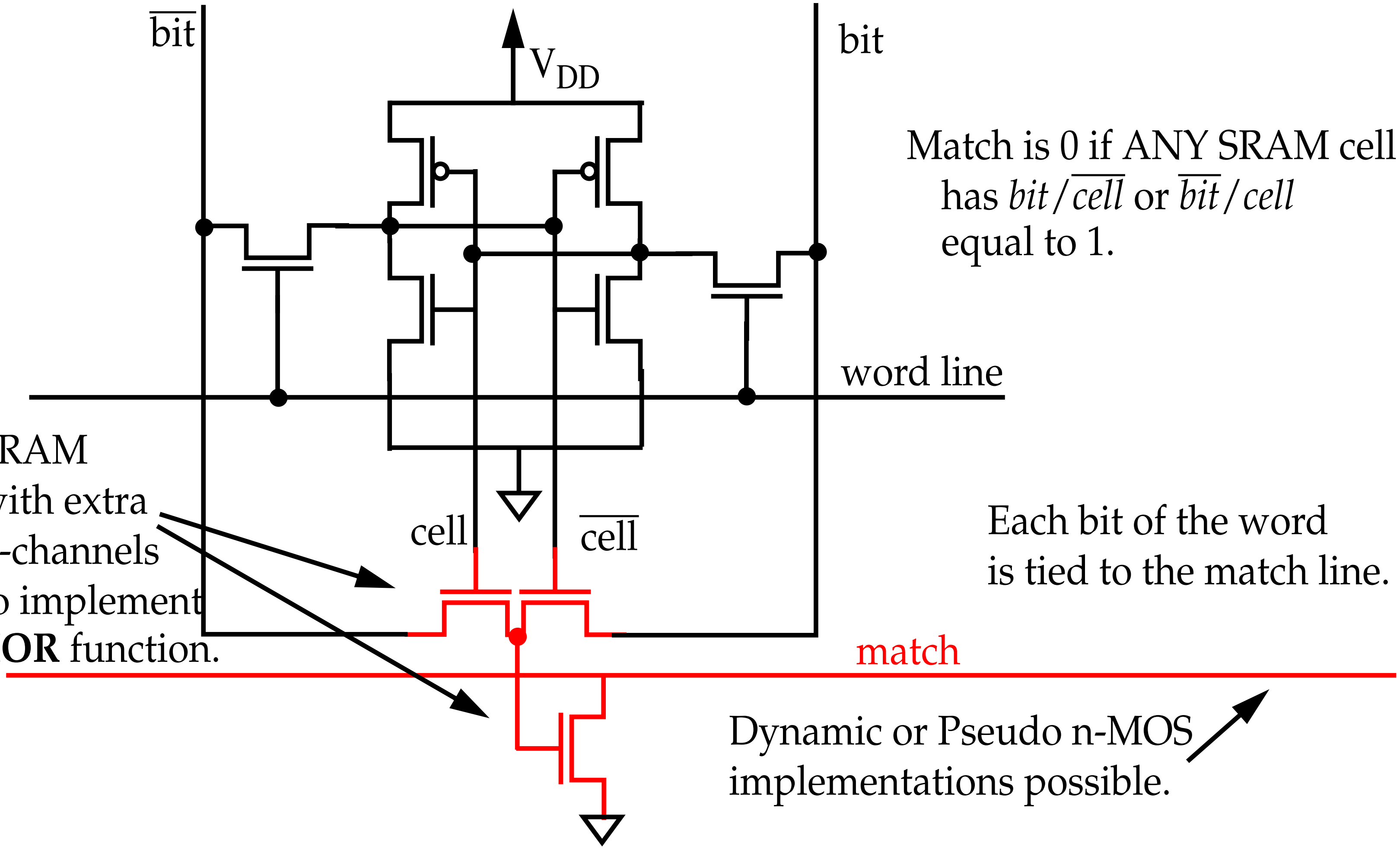
- 1T transistor requires an explicit capacitor (3T used gate capacitance). Capacitance must be large (~30fF) but area small - key challenge in design.
- Bootstrapping word-line to a value larger than V_{DD} circumvents V_T loss on storage capacitor.

Read-Write Memories (RAM)

Content Access Memory (CAM):

Determines if a match exists between a data word with a stored word.

Used in Translation-look-aside buffers.



Match is 0 if ANY SRAM cell has $bit/\overline{cell}$ or $\overline{bit}/cell$ equal to 1.

SRAM with extra n-channels to implement **XOR** function.

Each bit of the word is tied to the match line.

Dynamic or Pseudo n-MOS implementations possible.