ECE1388 VLSI Design Methodology

Lecture 13

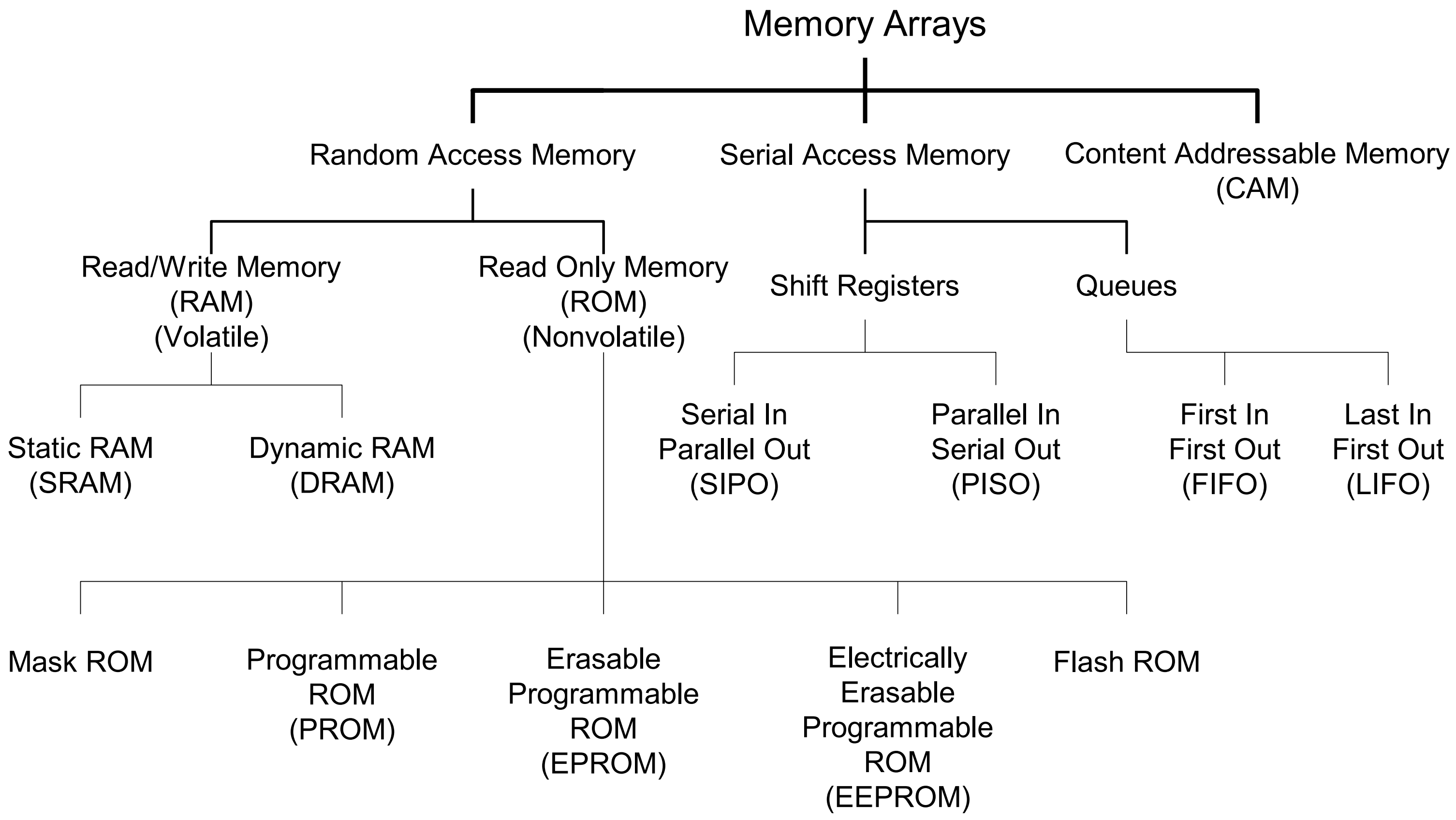
Memories I: SRAM

Adapted from Weste, Harris CMOS VLSI Design

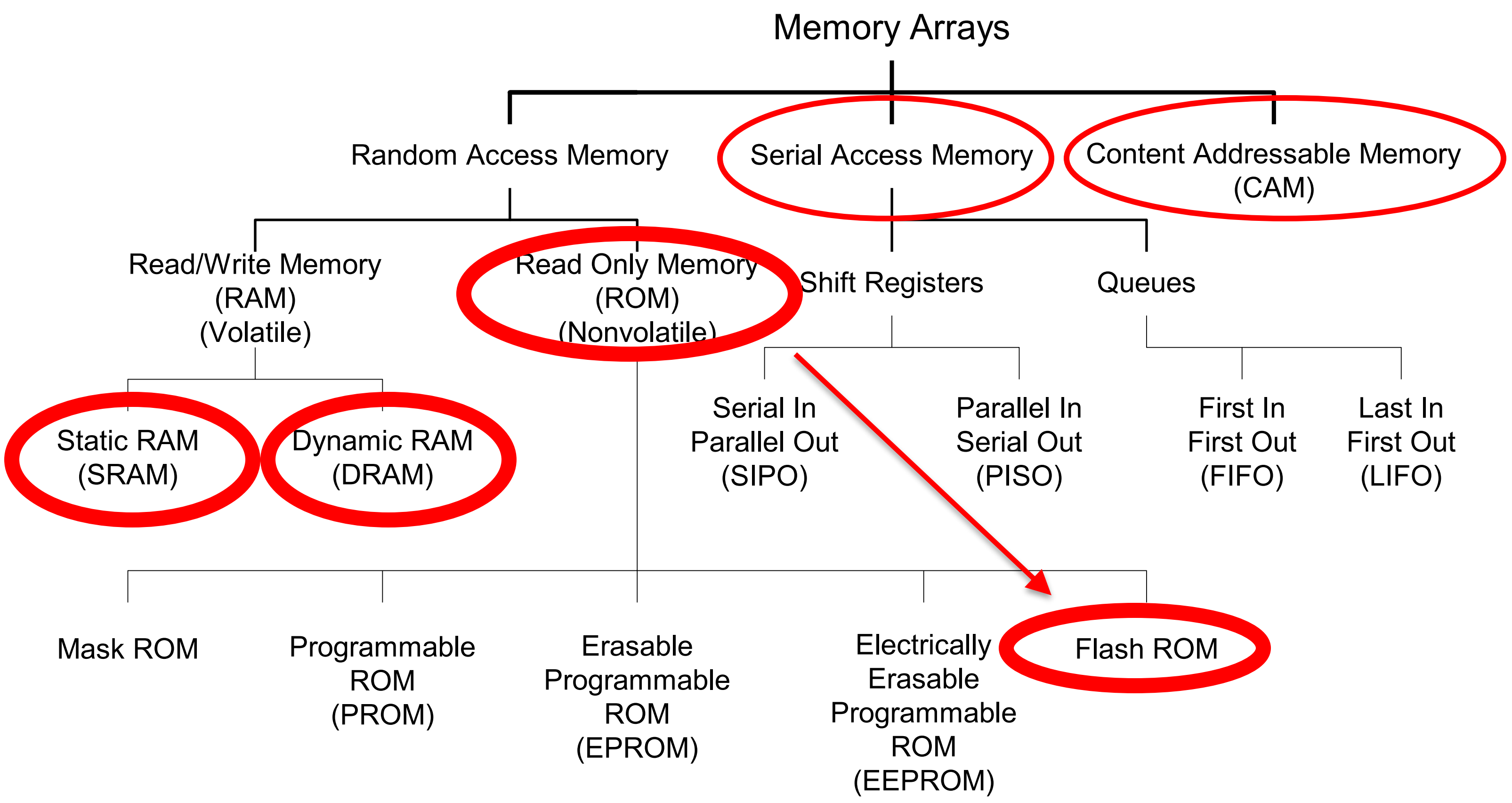
Outline

- Memory Arrays
- SRAM Architecture
  - SRAM Cell
  - Decoders
  - Column Circuitry
  - Multiple Ports

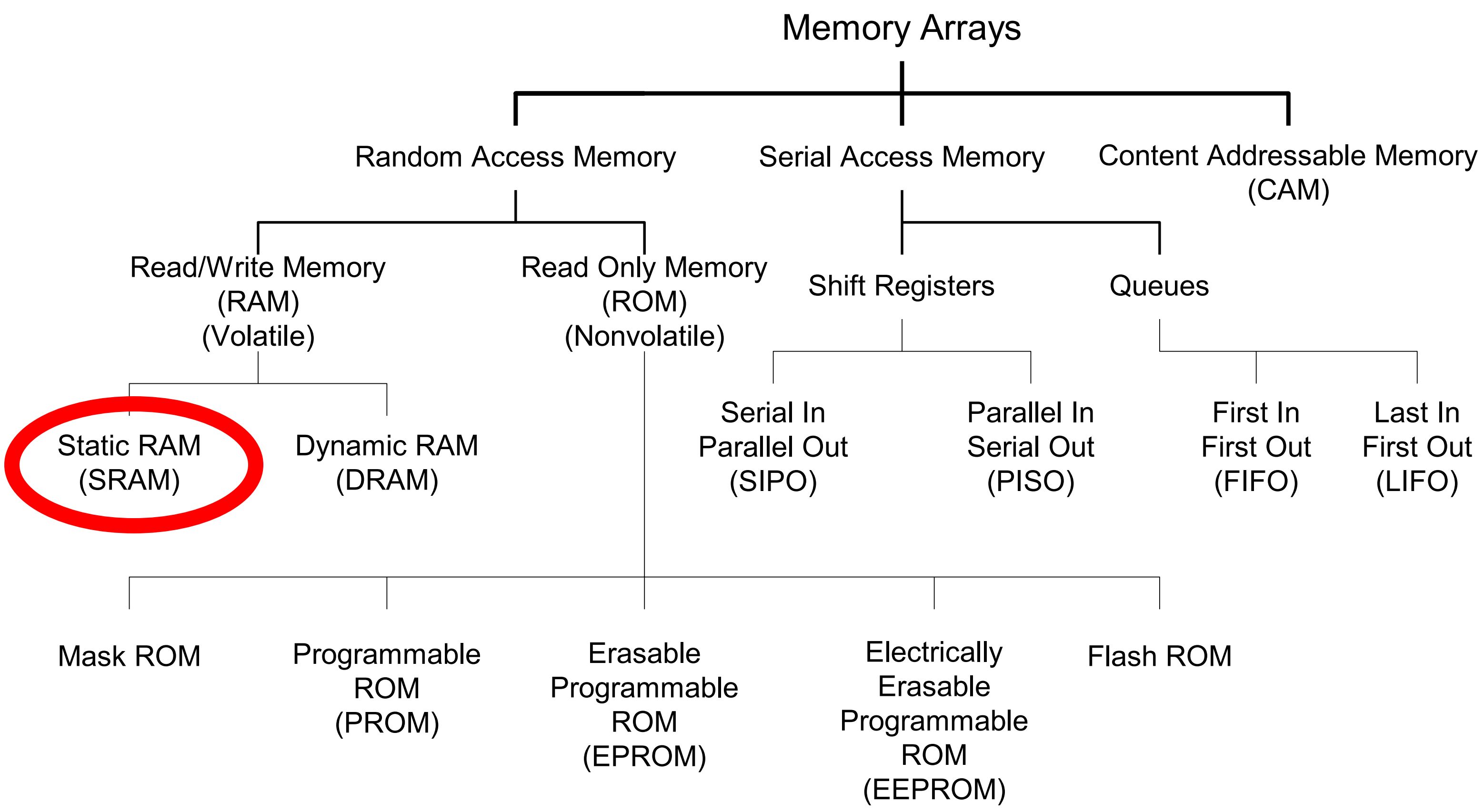
Memory Arrays


Memory Arrays
Random Access Memory
Serial Access Memory
Content Addressable Memory (CAM)
Read/Write Memory (RAM) (Volatile)
Read Only Memory (ROM) (Nonvolatile)
Shift Registers
Queues
Static RAM (SRAM)
Dynamic RAM (DRAM)
Serial In Parallel Out (SIPO)
Parallel In Serial Out (PISO)
First In First Out (FIFO)
Last In First Out (LIFO)
Mask ROM
Programmable ROM (PROM)
Erasable Programmable ROM (EPROM)
Electrically Erasable Programmable ROM (EEPROM)
Flash ROM

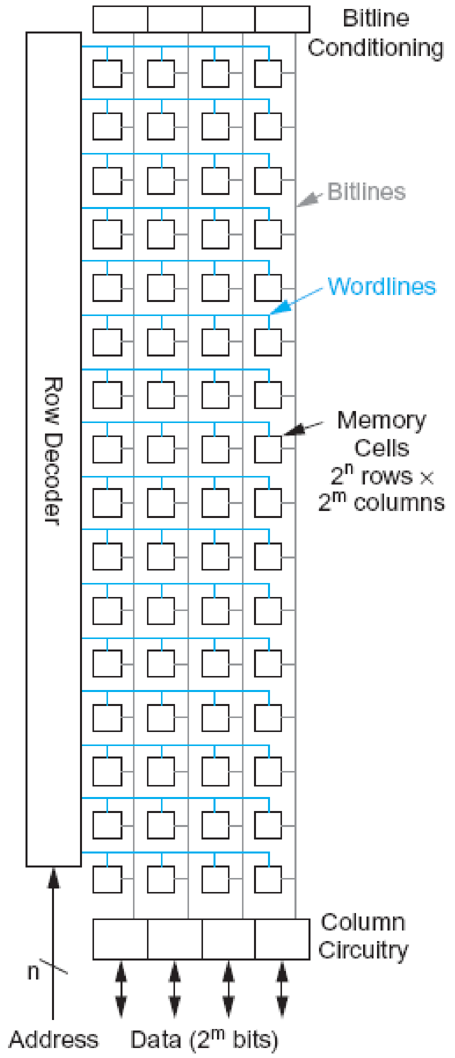
Memory Arrays

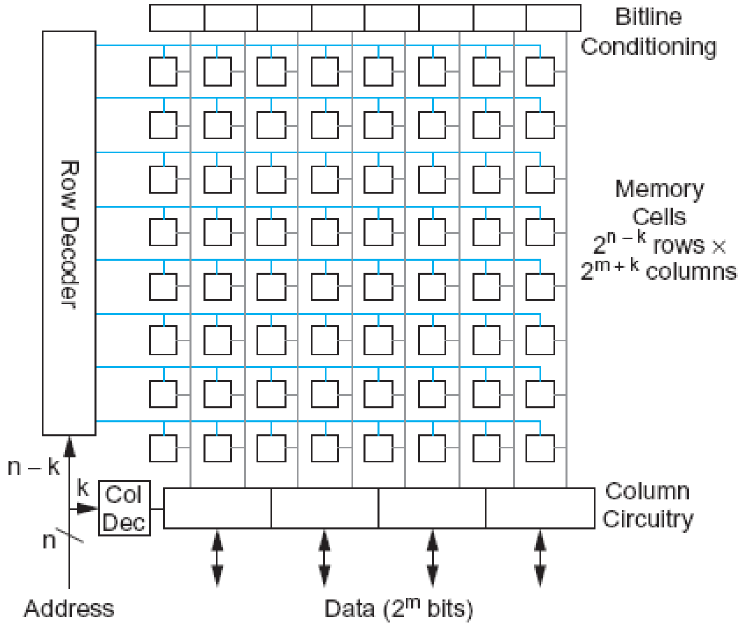

Memory Arrays
Random Access Memory
Serial Access Memory
Content Addressable Memory (CAM)
Read/Write Memory (RAM) (Volatile)
Read Only Memory (ROM) (Nonvolatile)
Shift Registers
Queues
Static RAM (SRAM)
Dynamic RAM (DRAM)
Serial In Parallel Out (SIPO)
Parallel In Serial Out (PISO)
First In First Out (FIFO)
Last In First Out (LIFO)
Mask ROM
Programmable ROM (PROM)
Erasable Programmable ROM (EPROM)
Electrically Erasable Programmable ROM (EEPROM)
Flash ROM

Memory Arrays


Memory Arrays
Random Access Memory
Serial Access Memory
Content Addressable Memory (CAM)
Read/Write Memory (RAM) (Volatile)
Read Only Memory (ROM) (Nonvolatile)
Shift Registers
Queues
Static RAM (SRAM)
Dynamic RAM (DRAM)
Serial In Parallel Out (SIPO)
Parallel In Serial Out (PISO)
First In First Out (FIFO)
Last In First Out (LIFO)
Mask ROM
Programmable ROM (PROM)
Erasable Programmable ROM (EPROM)
Electrically Erasable Programmable ROM (EEPROM)
Flash ROM

Array Architecture

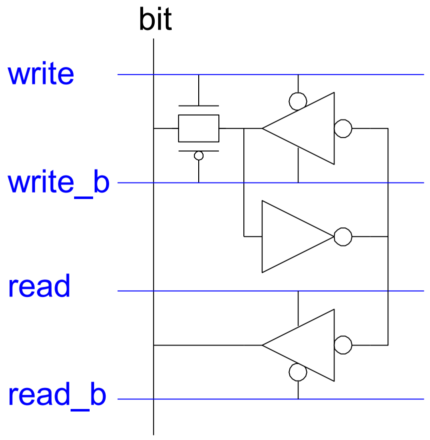


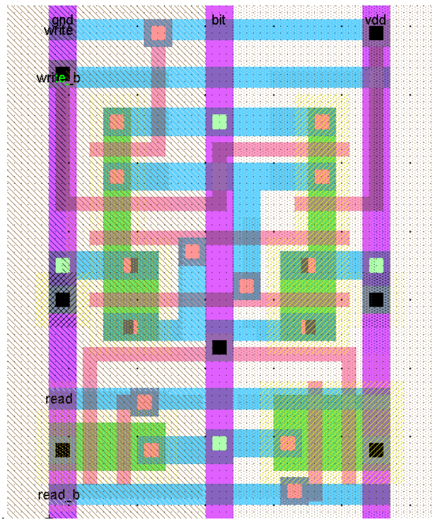


- 2^n *words* of 2^m *bits* each
- If n >> m, bit lines too long/slow
  - fold by 2^k into fewer *rows* of more *columns*
- Good regularity – easy to design
- Very high density if good cells are used

Brute Force: 12T SRAM Cell

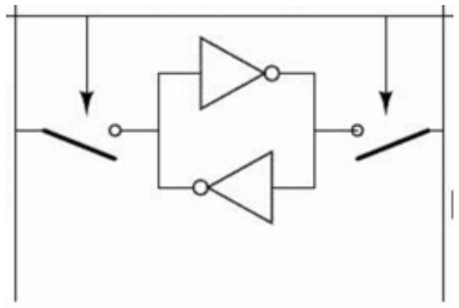
- Basic building block: SRAM Cell
  - Holds one bit of information, like a latch
  - Must be read and written
- *Brute force approach*: 12-transistor (12T) SRAM cell
  - Use a simple latch connected to bitline
  - 46 x 75 λ unit cell – very large

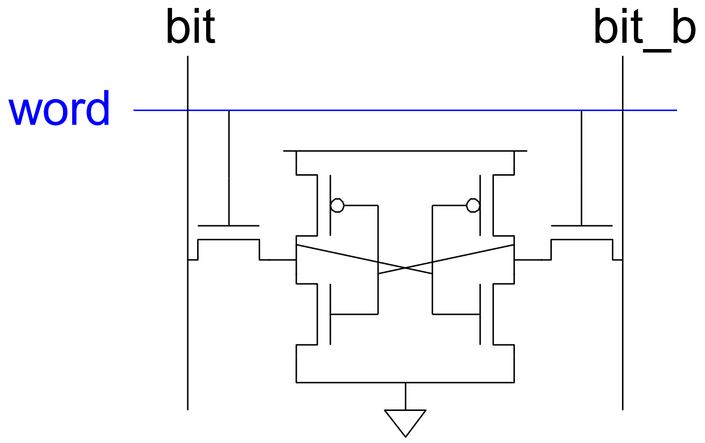




6T SRAM Cell

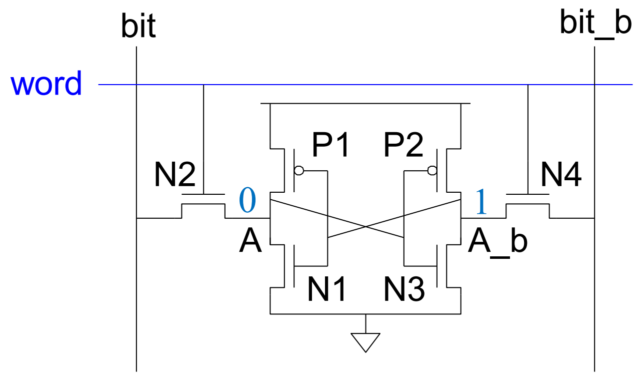
- ❑ Cell size accounts for most of array size
  - Reduce cell size at expense of complexity
- ❑ 6T SRAM Cell
  - Used in most commercial chips
  - Data stored in cross-coupled inverters
- ❑ Read:
  - Precharge bit, bit_b
  - Raise wordline
- ❑ Write:
  - Drive data onto bit, bit_b
  - Raise wordline

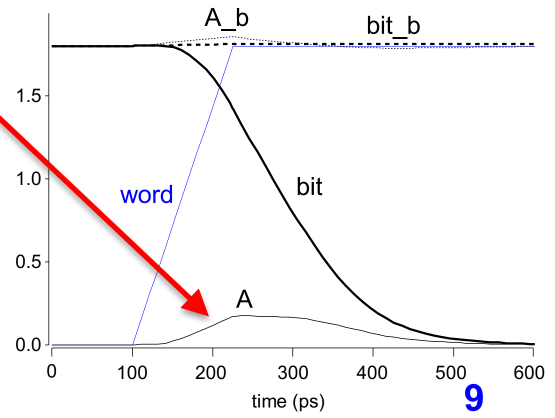




SRAM Read

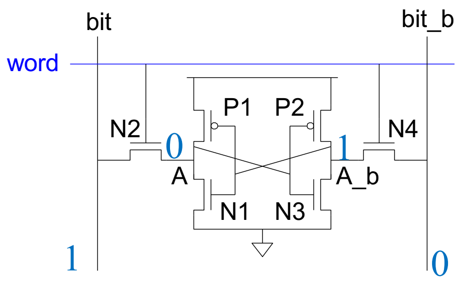
- Precharge both bitlines high
- Then turn on wordline
- One of the two bitlines will be pulled down by the cell
- Ex: A = 0, A_b = 1
  - bit discharges, bit_b stays high
  - But A bumps up slightly
- *Read stability*
  - A must not flip
  - N1 >> N2 (& N3 >> N4)
    - N2-current source (saturation)
    - N1-resistor (triode)

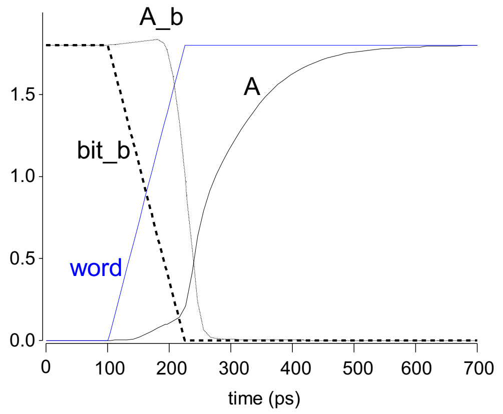




SRAM Write

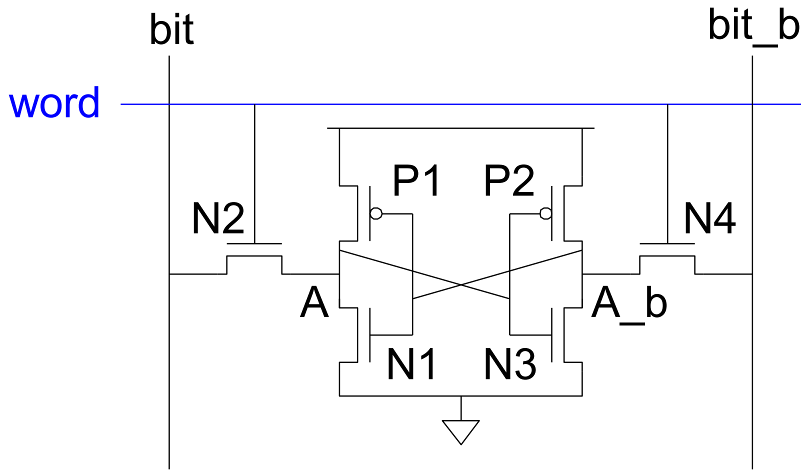
- Drive one bitline high, the other low
- Then turn on wordline
- Bitlines overpower cell with new value
- Ex: A = 0, A_b = 1, bit = 1, bit_b = 0
  - Force A_b low, then A rises high
- *Writability*
  - Must overpower feedback inverter
  - N4 >> P2 (& N2 >> P1)
    - N4 initially a current source
    - P2 initially a resistor
    - As A_b drops to Vtn these roles swap

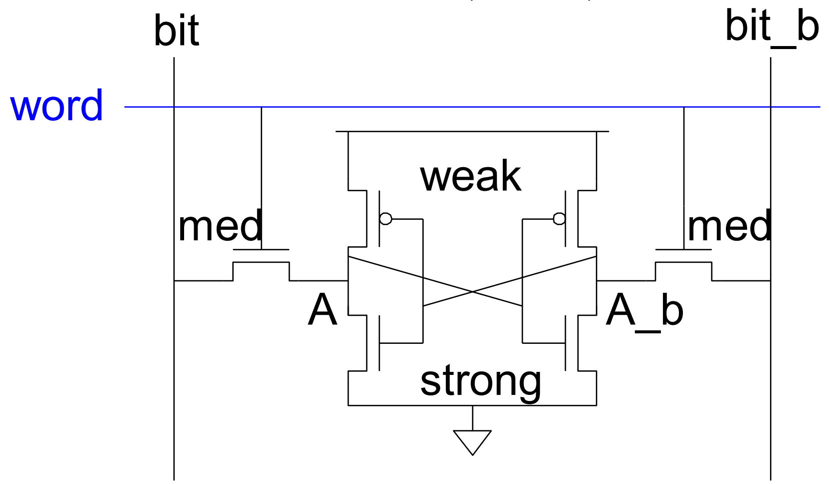




SRAM Sizing

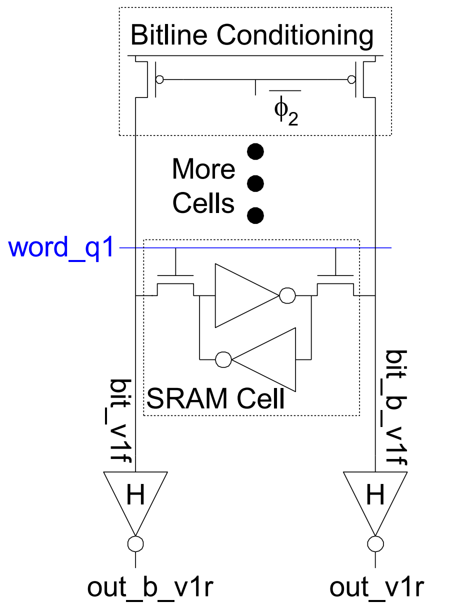
- Three unknowns (N1, N2, P1) and three constraints:
  1. **Read stability**: High bitlines must not overpower inverters during reads: *N1 "stronger" than N2*
  2. **Writability**: But low bitlines must write new value into cell: *N2 "stronger" than P1*
  3. **Read speed**: dictated by size of *N2 within a range*
- Write out three equations and solve for N1, N2, P1



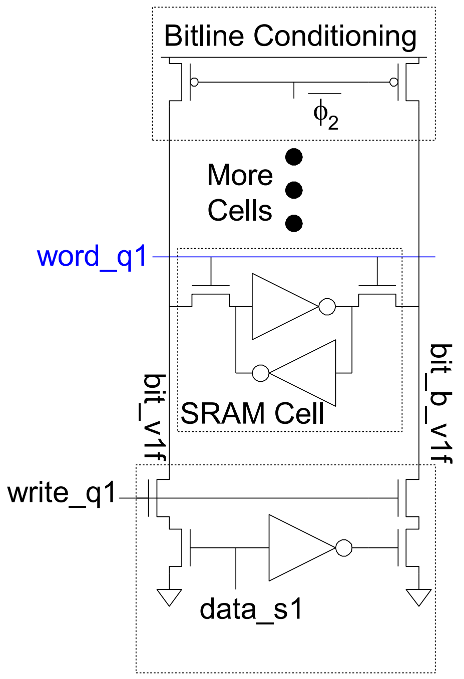


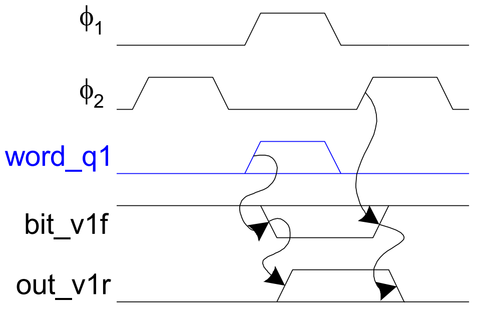
SRAM Column Example

Read



Write

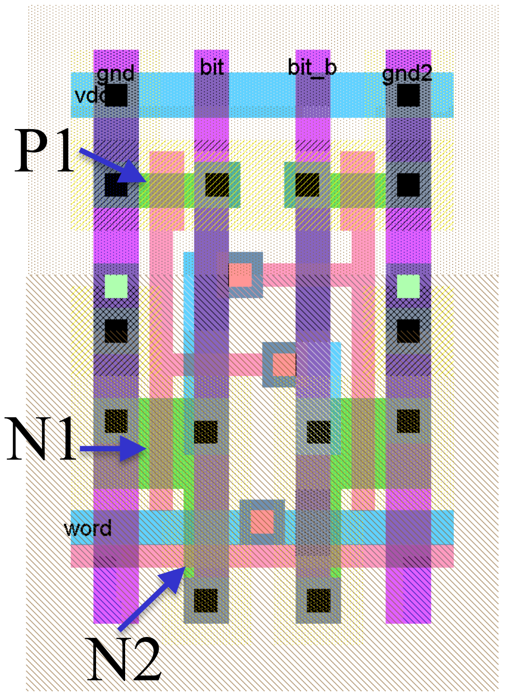


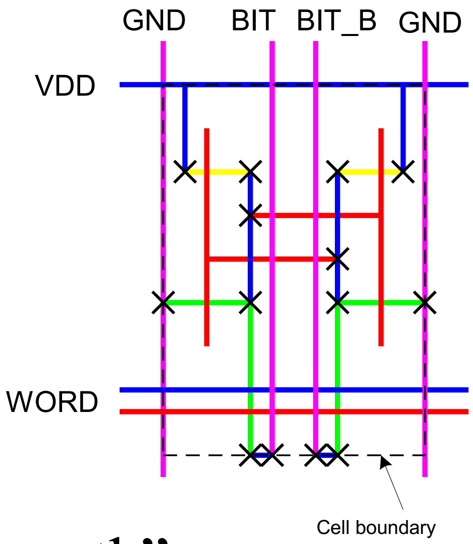


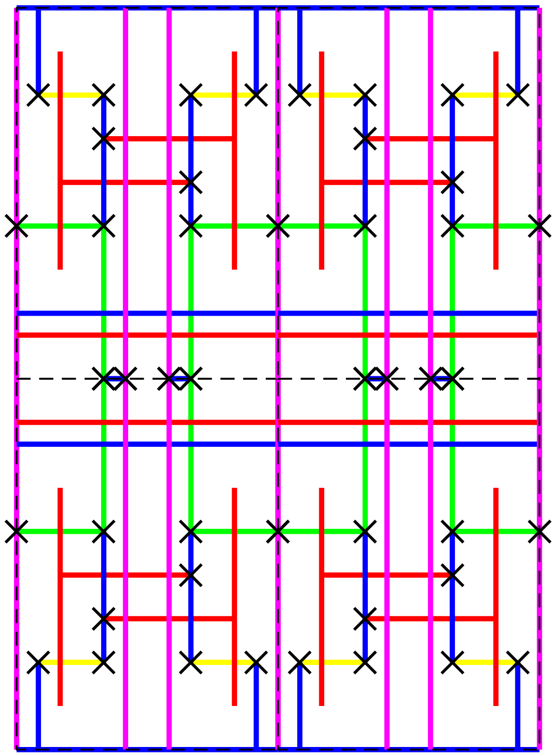
- Large-signal read
  - For small arrays
- Need a differential “sense” amplifier
  - To speed up for large arrays
  - To add tolerance to CM noise

SRAM Layout

- Cell size is critical: 26 x 45 λ (even smaller in industry)
- Tile cells sharing V_{DD}, GND, bitline contacts



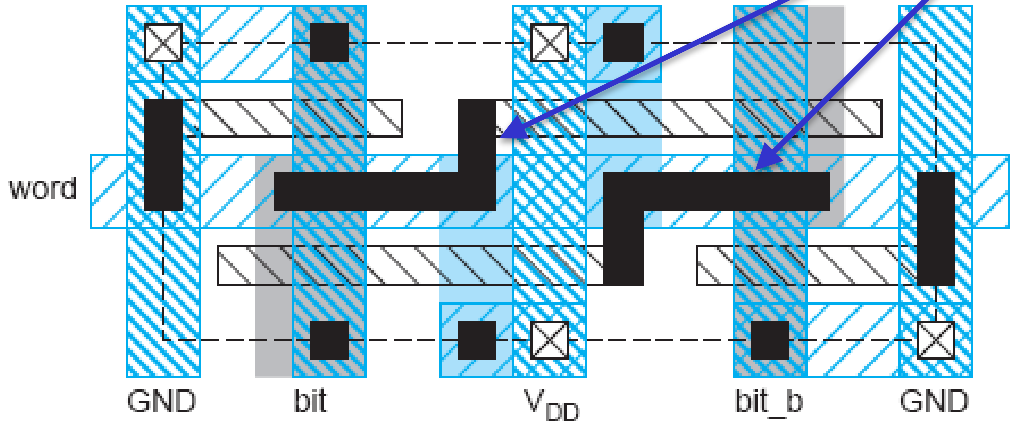




N1 >> N2 >> P1 wrt "strength"

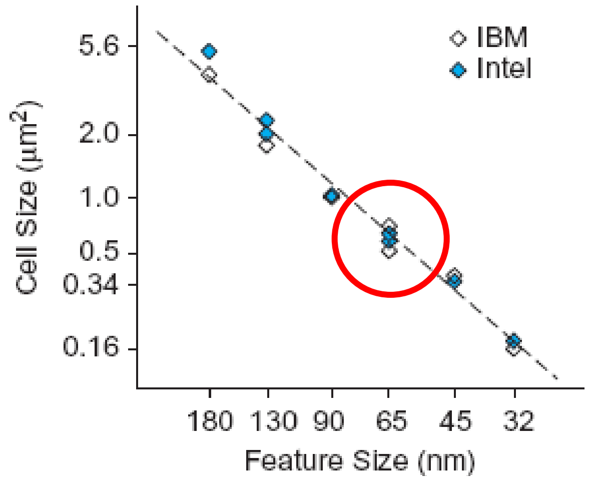
6T Thin Cell

- In nanometer CMOS
  - Avoid bends in polysilicon and diffusion
  - Orient all transistors in one direction
- *Lithographically friendly* or *thin-cell* layout does this
  - Diffusion – only vertical; Poly – only horizontal
  - Uses local interconnect with trench contacts
  - Also reduces length and capacitance of bitlines

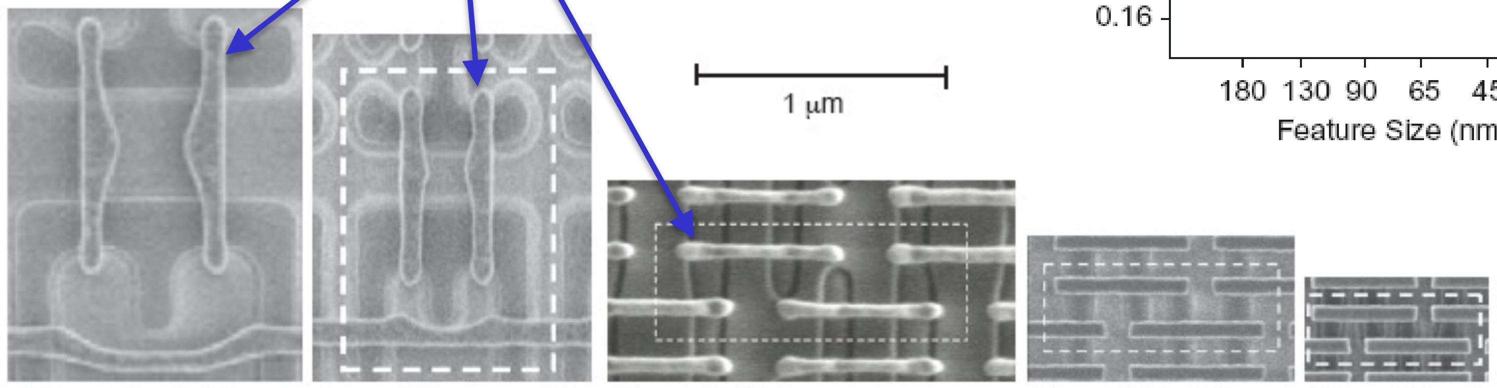


Commercial SRAMs

- Five generations of Intel SRAM cell micrographs
  - Transition to thin cell at 65 nm
  - Steady scaling of cell area



No bends in polysilicon



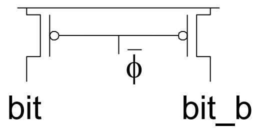
130 nm [Tyagi00] 90 nm [Thompson02] 65 nm [Bai04] 45 nm [Mistry07] 32 nm [Natarajan08]

Column Circuitry

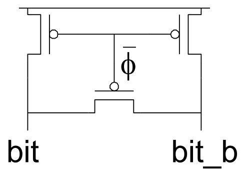
- Some circuitry is required for each column
  - Bitline conditioning
  - Sense amplifiers
  - Column multiplexing

Bitline Conditioning

- Precharge bitlines high before reads



- Equalize bitlines to minimize voltage difference when using sense amplifiers

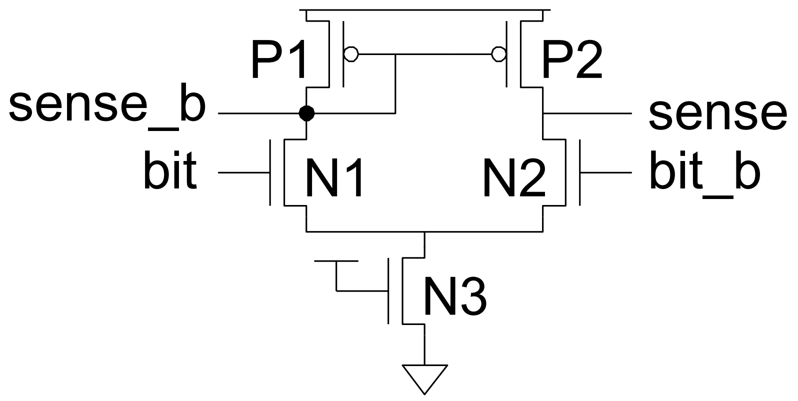


Sense Amplifiers

- Bitlines have many cells attached
  - Ex: 32-kbit SRAM has 128 rows x 256 cols
  - 128 cells on each bitline
- $t_{pd} \propto (C/I)\,\Delta V$
  - Even with shared diffusion contacts, 64C of diffusion capacitance (large C)
  - Discharged slowly through small transistors (small I)
- *Sense amplifiers* are triggered on small voltage swing (reduce ΔV)

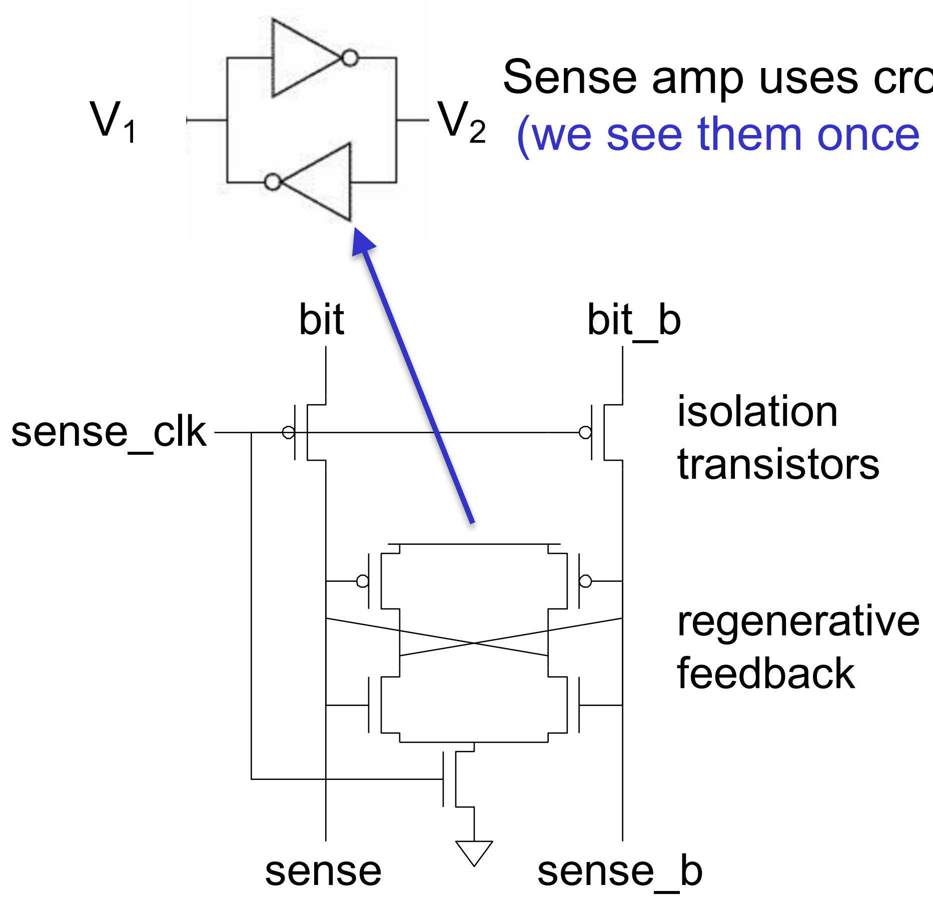
Brute Force: Differential Pair Amp

- Differential pair requires no clock
- But always dissipates static power – not a good choice

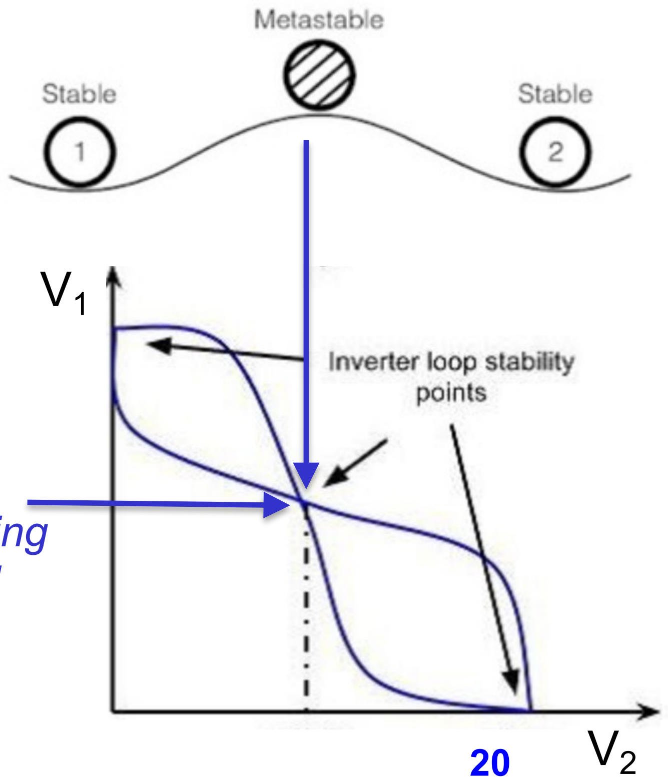


Clocked Sense Amp

- Clocked sense amp saves power
- Requires sense_clk after enough bitline swing
- Isolation transistors cut off large bitline capacitance



Sense amp uses cross-coupled inverters
(we see them once again – very useful positive-feedback circuit!)



Metastable point can be near Vdd using high-skewed inverters

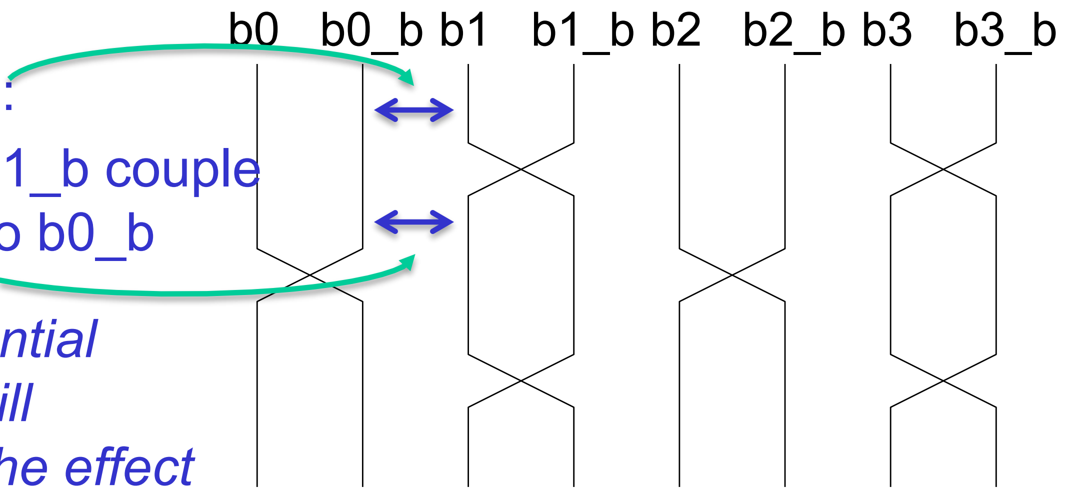
Twisted Bitlines

- Sense amplifiers also amplify noise
  - Coupling noise is severe in modern processes
  - Try to couple equally onto bit and bit_b
  - Done by *twisting* bitlines

Example:

b1 and b1_b couple equally to b0_b

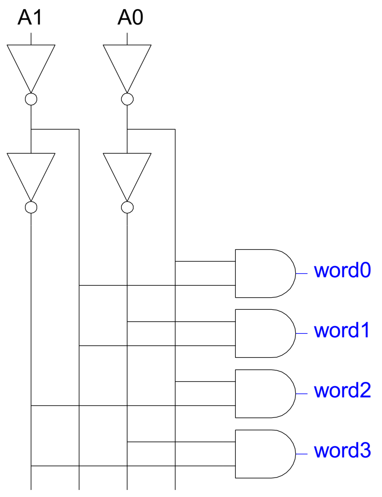
So the differential sense amp will subtract out the effect of coupling



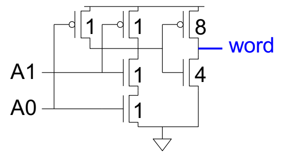
Decoders

- n:2^n decoder consists of 2^n n-input AND gates
  - One needed for each row of memory
  - Build AND from NAND or NOR gates

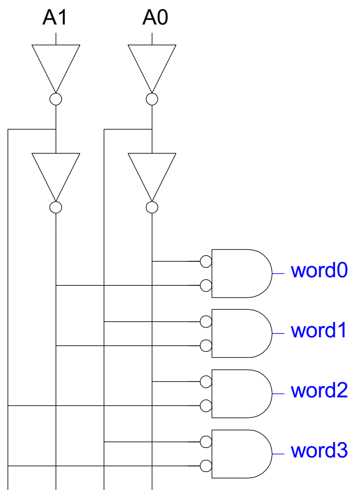
Static CMOS



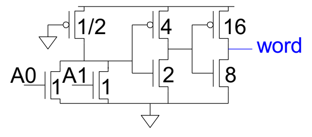
NAND + INV



Pseudo-nMOS (power too high)

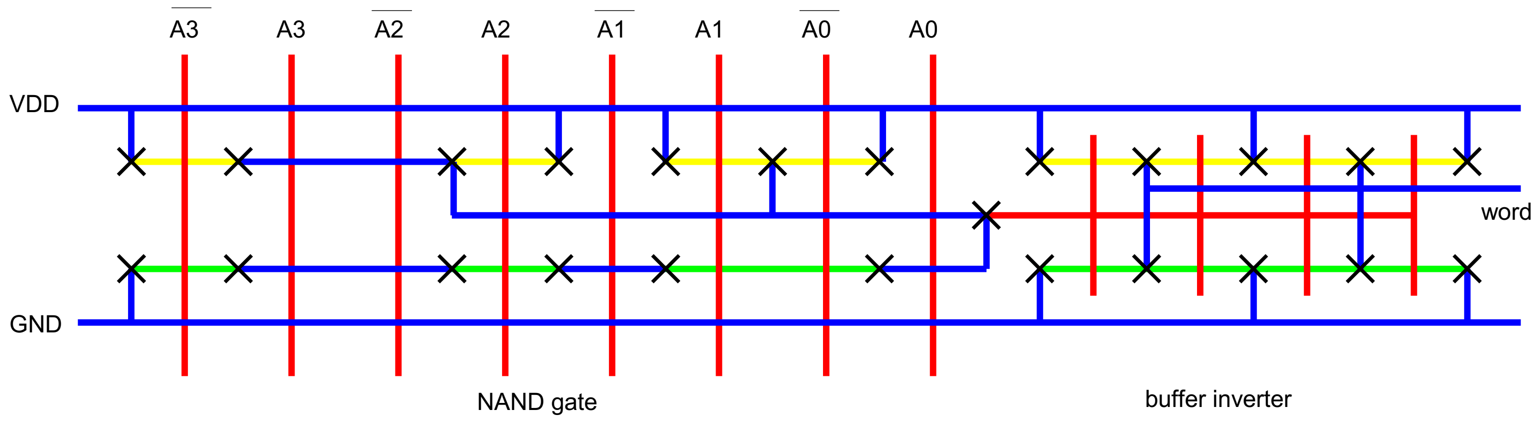


NOR + INV + INV



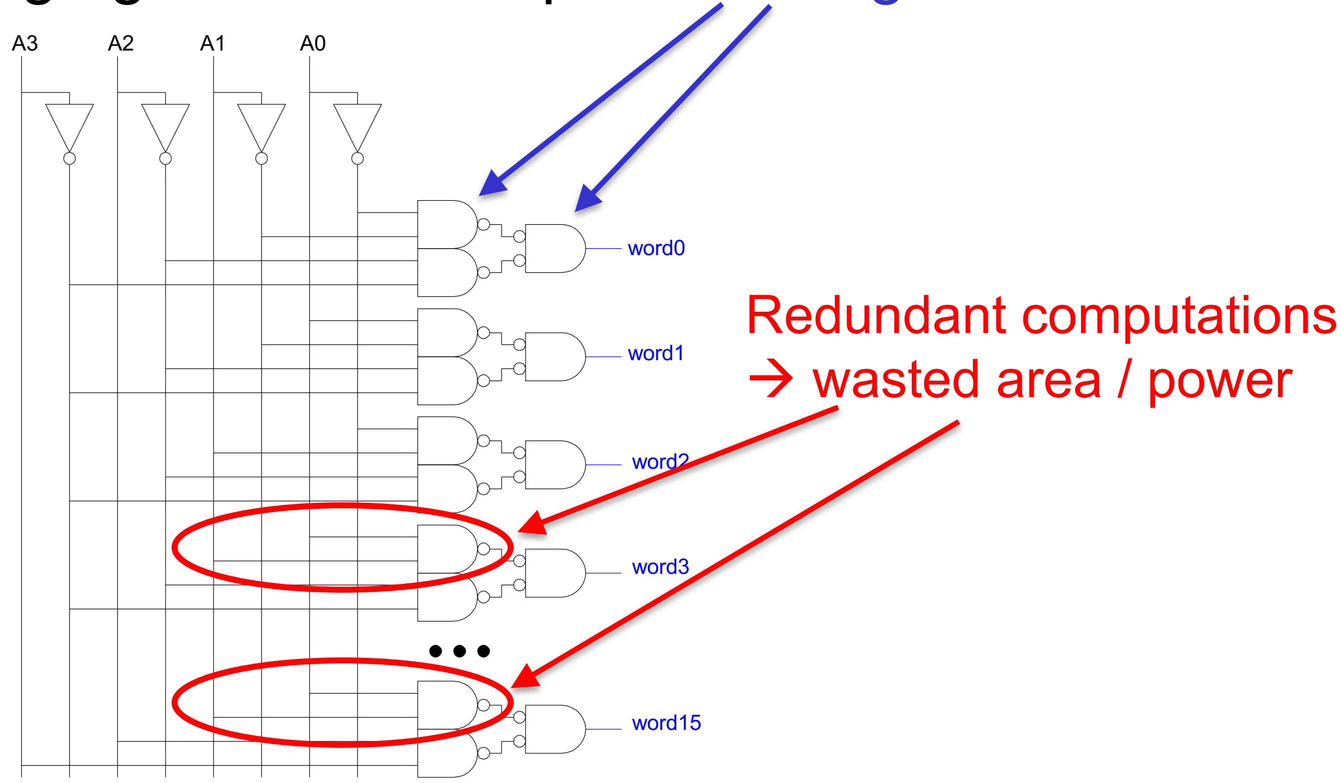
Decoder Layout

- Decoders must be pitch-matched to SRAM cell
  - Requires very skinny gates



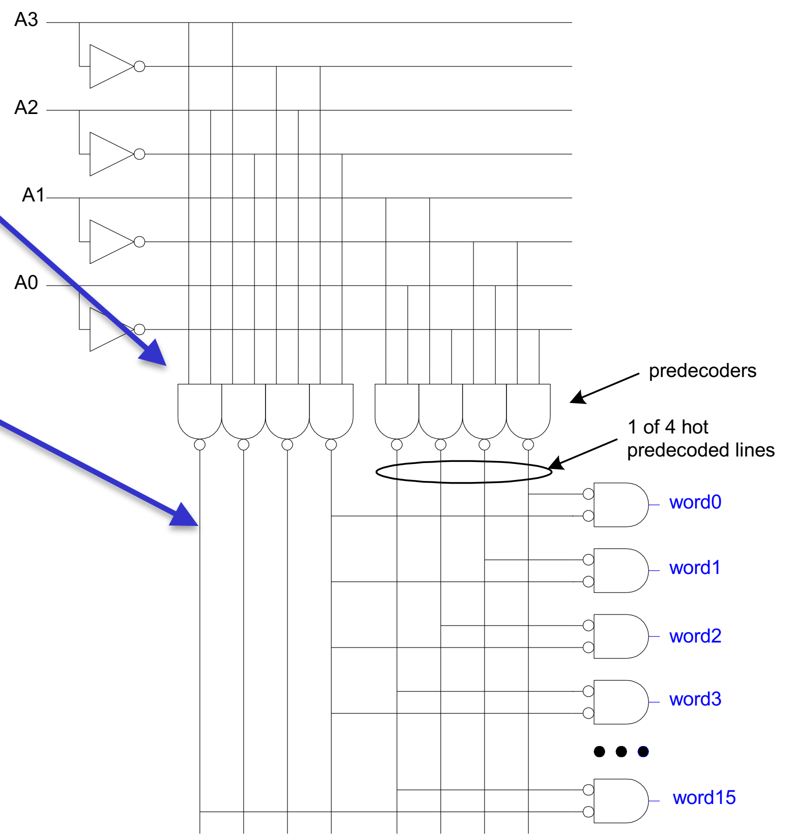
Large Decoders

- For n > 4, NAND gates become slow
  - Break large gates into multiple smaller gates



Predecoding

- Many of these gates are redundant (see previous slide)
  - Factor out common gates into predecoder
  - And "broadcast" partial results vertically (instead of address bits)
  - Saves area
  - Same path effort

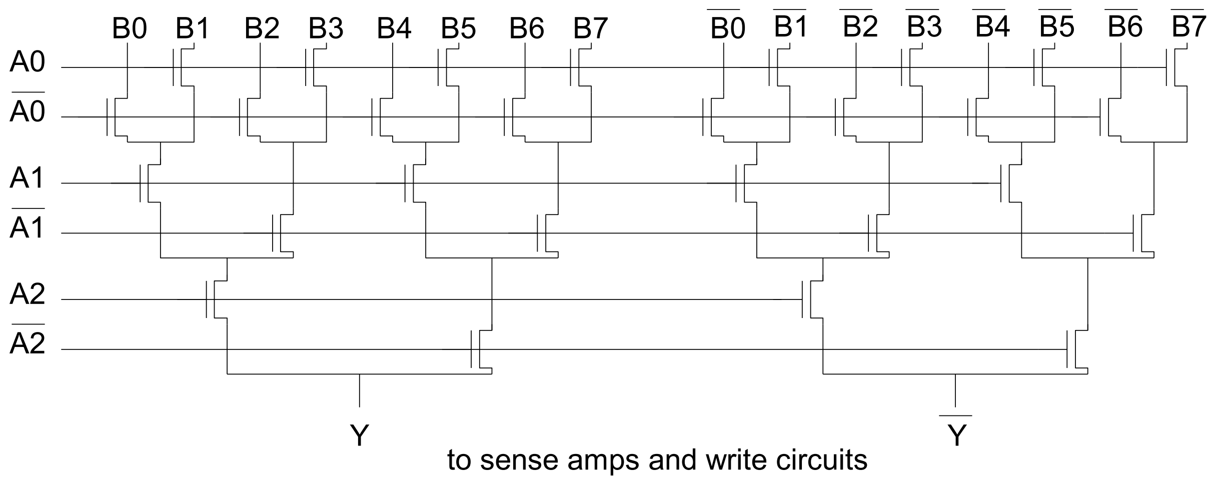


Column Multiplexing

- Recall that array may be folded for good aspect ratio
- Ex: 2 kword x 16 folded into 256 rows x 128 columns
  - Must select 16 output bits from the 128 columns
  - Requires 16 8:1 column multiplexers

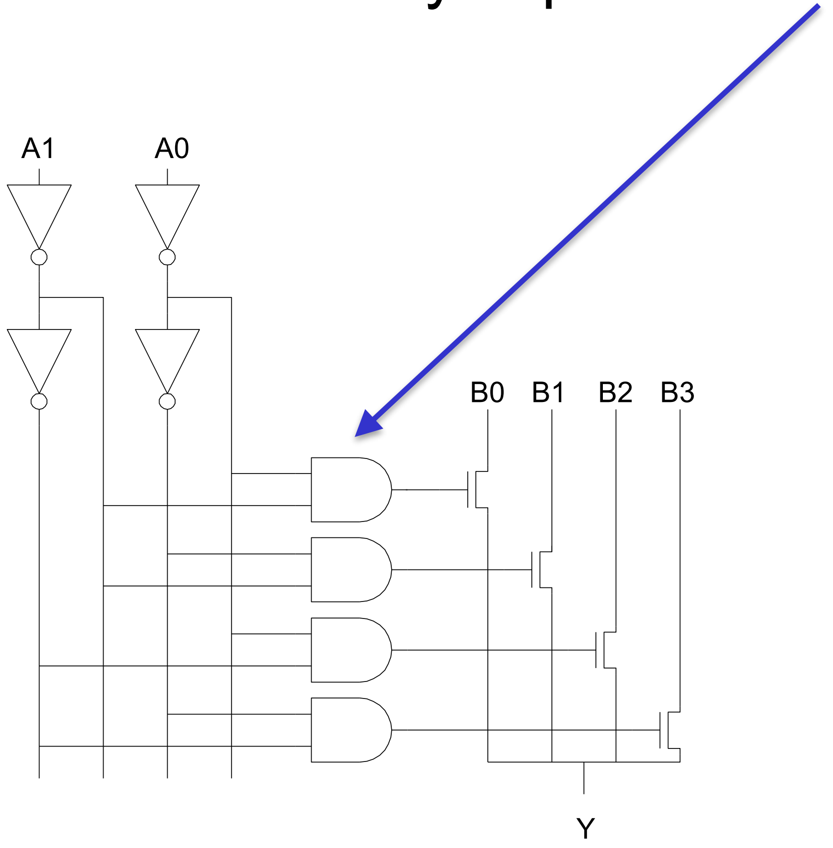
Tree Decoder Mux

- Column mux can use pass transistors
  - Use nMOS only, precharge outputs
- One design is to use k series transistors for 2^k:1 mux
  - No external decoder logic needed, but is slow (RC ladder)



Single Pass-Gate Mux

- Eliminate series transistors by separate decoder to speed up

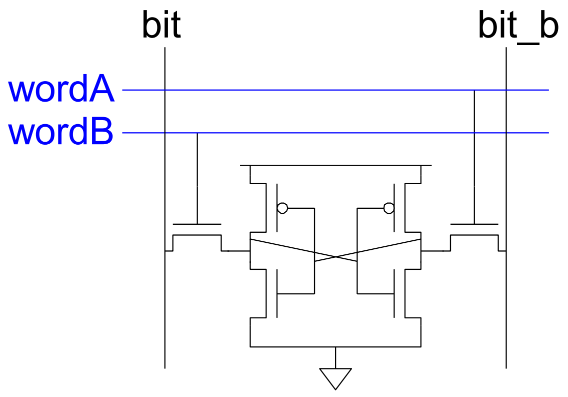


Multiple Ports

- We have considered single-ported SRAM
  - One read or one write on each cycle
- *Multiported* SRAM are needed for register files
- Examples:
  - Pipelined MIPS must read two sources and write a third result each cycle
  - Multicycle MIPS must read two sources or write a result on some cycles
  - Superscalar MIPS must read and write many sources and results each cycle

Dual-Ported SRAM

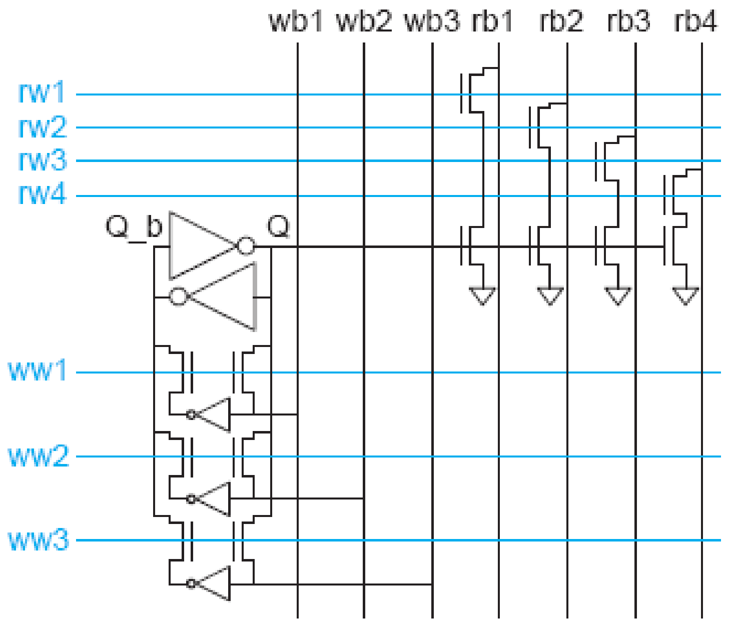
- Simple dual-ported SRAM
  - Two independent single-ended reads
  - Or one differential write



- Do two reads and one write by time multiplexing
  - Read during ph1, write during ph2

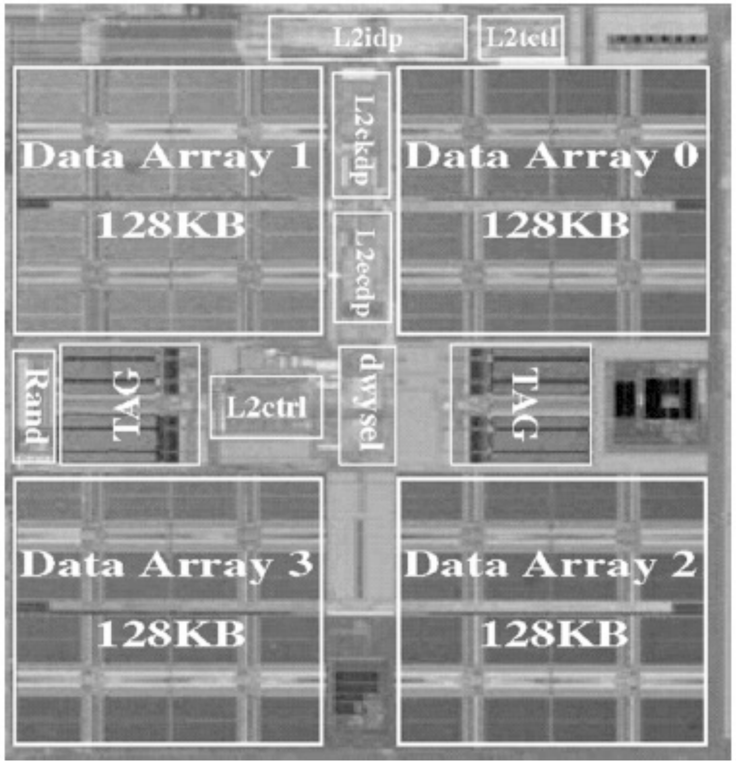
Multi-Ported SRAM

- Adding more access transistors hurts read stability
- Multiported SRAM isolates reads from state node
- Single-ended bitlines save area



Large SRAMs

- Large SRAMs are split into subarrays for speed
- Ex: UltraSparc 512KB cache
  - 4 128 KB subarrays
  - Each have 16 8KB banks
  - 256 rows x 256 cols / bank
  - 60% subarray area efficiency
  - Also space for tags & control



[Shin05]