University of
Massachusetts
Amherst

ECE124
Intro-Digital and Computer Sys

Chapter 7

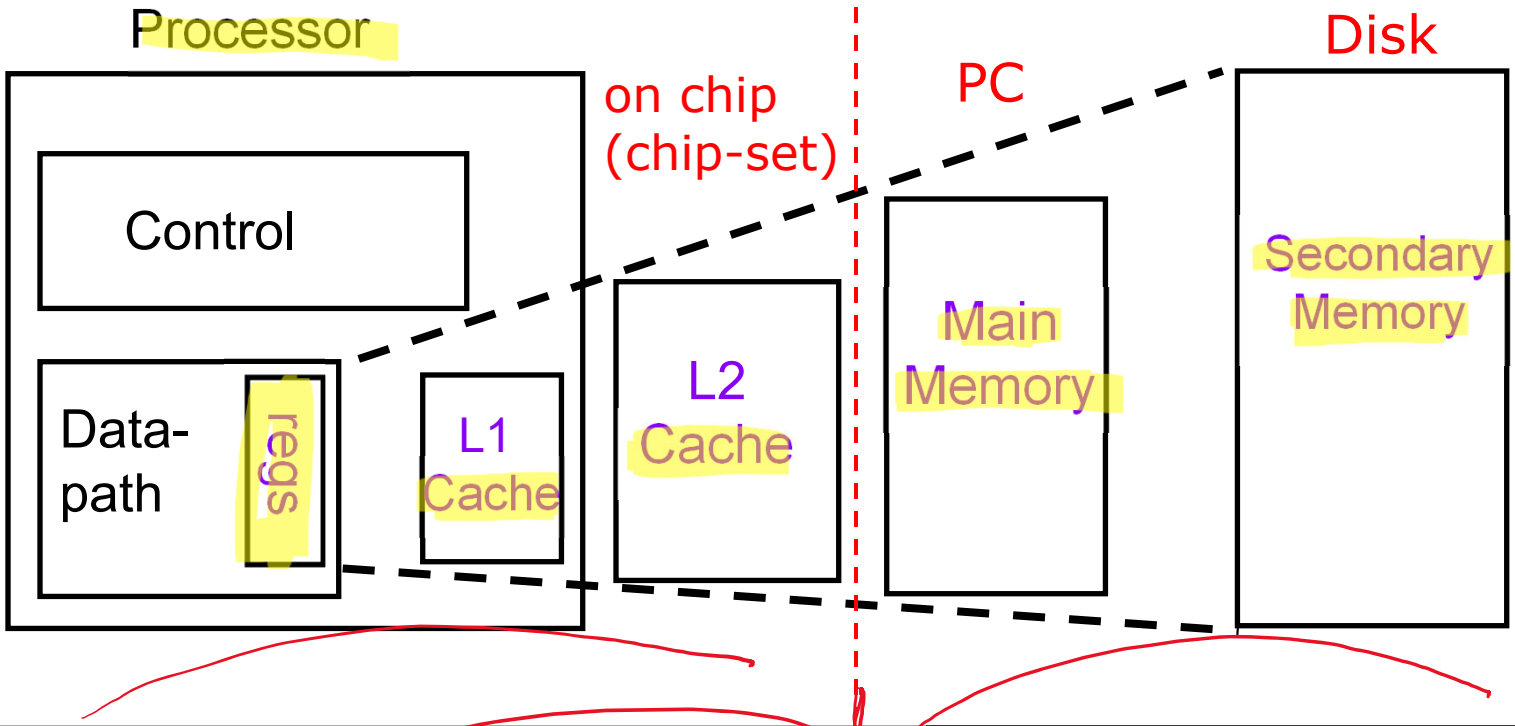
Memory and Programmable Logic

Memories

- How is memory used in computers?
  - Memory hierarchy (next slide)
- What differentiates various types of memories?
  - Technology: magnetic, charge, capacitance, resistance, fuse
  - Access Flexibility: random (access anything), sequential (access in order)
    - » Random Access Memory (RAM) vs. Sequential Access Memory (SAM)
    - » USB memory vs. Magnetic type memory
    - » Access/Cycle Time: time required to complete a read/write operation
  - Access Options (Memory function): Write (i.e. store) and read (i.e. retrieve), or read only
    - » Read-Write Memory (RWM) vs. ROM (Read-Only Memory)
  - Volatility: non-volatile or volatile (does or does not retain data on power-down)
  - Capacity: amount of data that can be stored—typically expressed in bytes
    - » Word Size: # bits stored or retrieved upon access
- We'll consider two conventional memories
  - Random-Access Memory (RAM)
  - Read-Only Memory (ROM)
- Is memory combinational or sequential logic?
  - RAM: array of binary storage elements like sequential, but not clocked (latch)
  - ROM: combinational (decoder + OR gates)

AND NAND gate

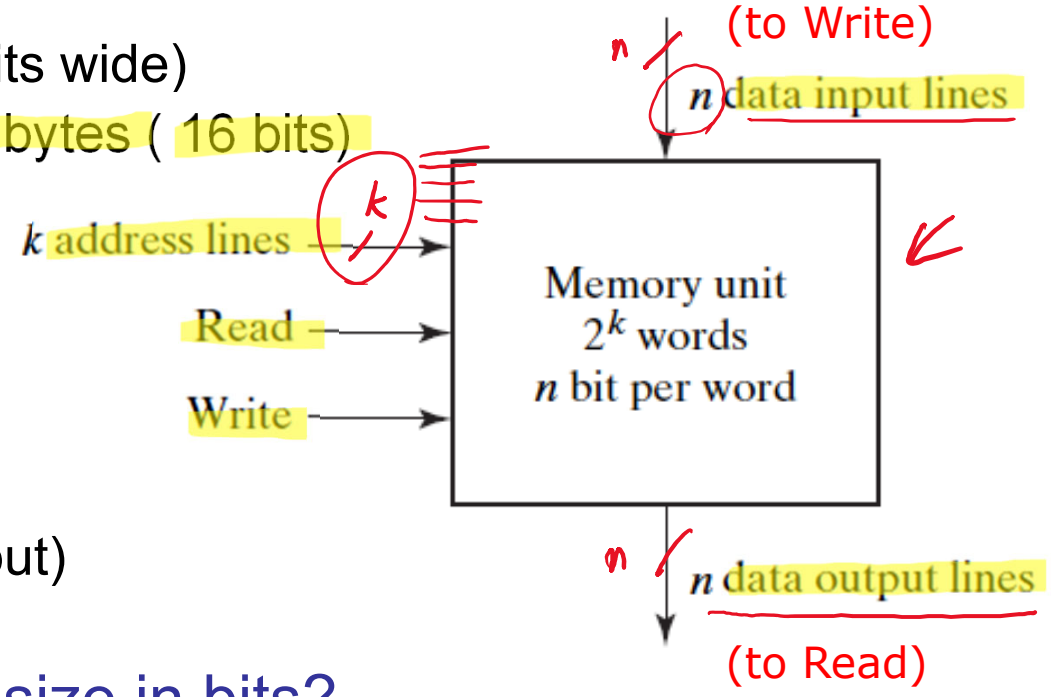
Memory Hierarchy



Speed (ns):	0.1ns	0.2ns	1ns	6ns	3,000,000ns
Size (MB):	0.00	0.5	2-6	4,000-8,000+	500,000+
Cost ($/MB):	--	--	$1	$0.004	$0.00001
Technology:	Regs	SRAM	SRAM	DRAM	Disk

Random-access Memory (RAM)

- Memory terminology:
  - Smallest unit is "word" (*n* bits wide)
  - 1 byte is 8 bits, 1 word is 2 bytes (16 bits)
  - Identified by "address"



- Block diagram:
  - *k* address lines
    - » At most 2^k words storage
  - *n* data lines (input and output)

- What is the total memory size in bits?
  - Size = $2^k \times n$ bits

- Example:
  - 1024-bit memory with 16-bit words
  - How many address lines?

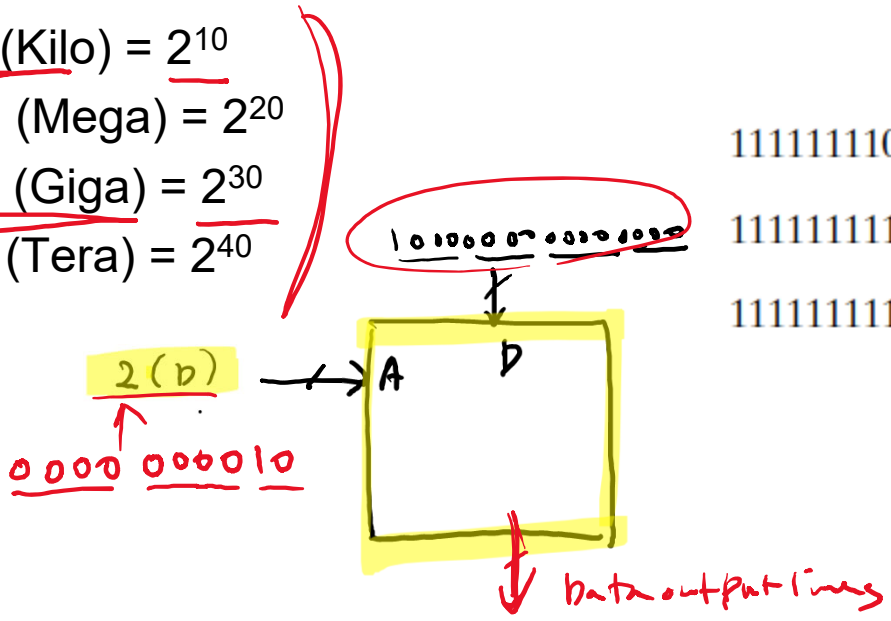
Memory Layout

- **Example: 1024x16 RAM**
  - 1024 16-bit words
  - 10 bits identify address (or "location") of word
  - 16 bit word stored at each location

- Remember:
  - k (Kilo) = 2^{10}
  - M (Mega) = 2^{20}
  - G (Giga) = 2^{30}
  - T (Tera) = 2^{40}

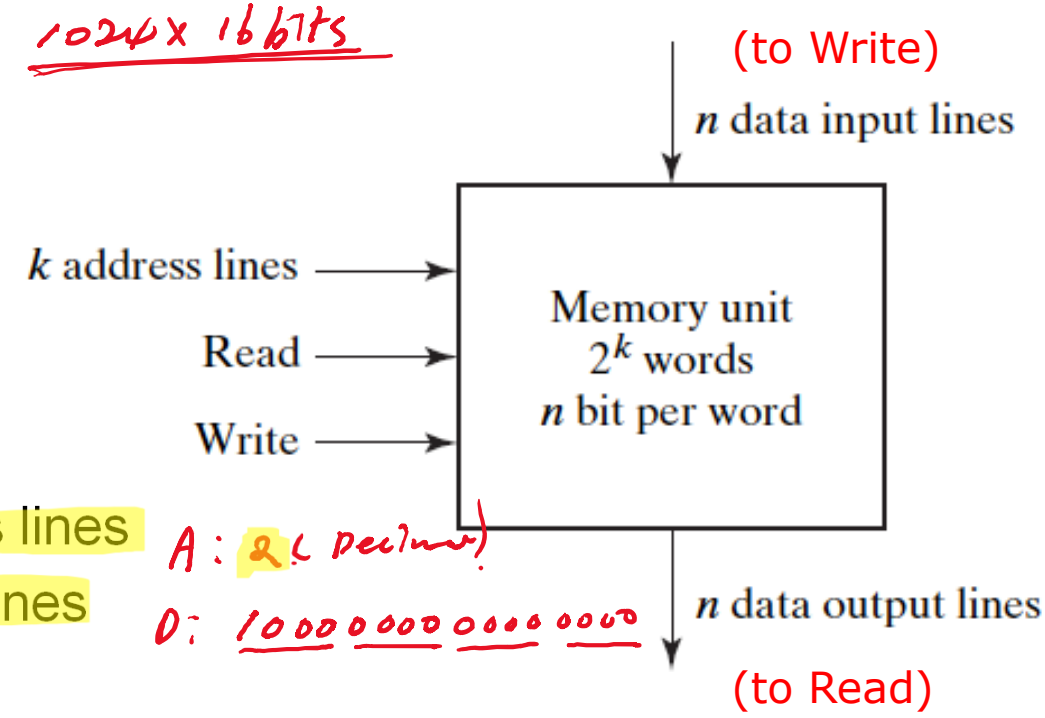
Memory address

Binary	decimal	Memory content
0000000000	0	1011010101011101
0000000001	1	1010101110001001
0000000010	2	0000110101000110
		1000000000010000
1111111101	1021	1001110100010100
1111111110	1022	0000110100011110
1111111111	1023	1101111000100101



RAM Read and Write

- Control inputs:
  - Address
  - Read
  - Write



- Write operation
  1. Apply address to address lines
  2. Apply data to data input lines
  3. Activate write input

- Read operation
  1. Apply address to address line
  2. Activate read input
  3. Read value from data output lines

RAM Read and Write

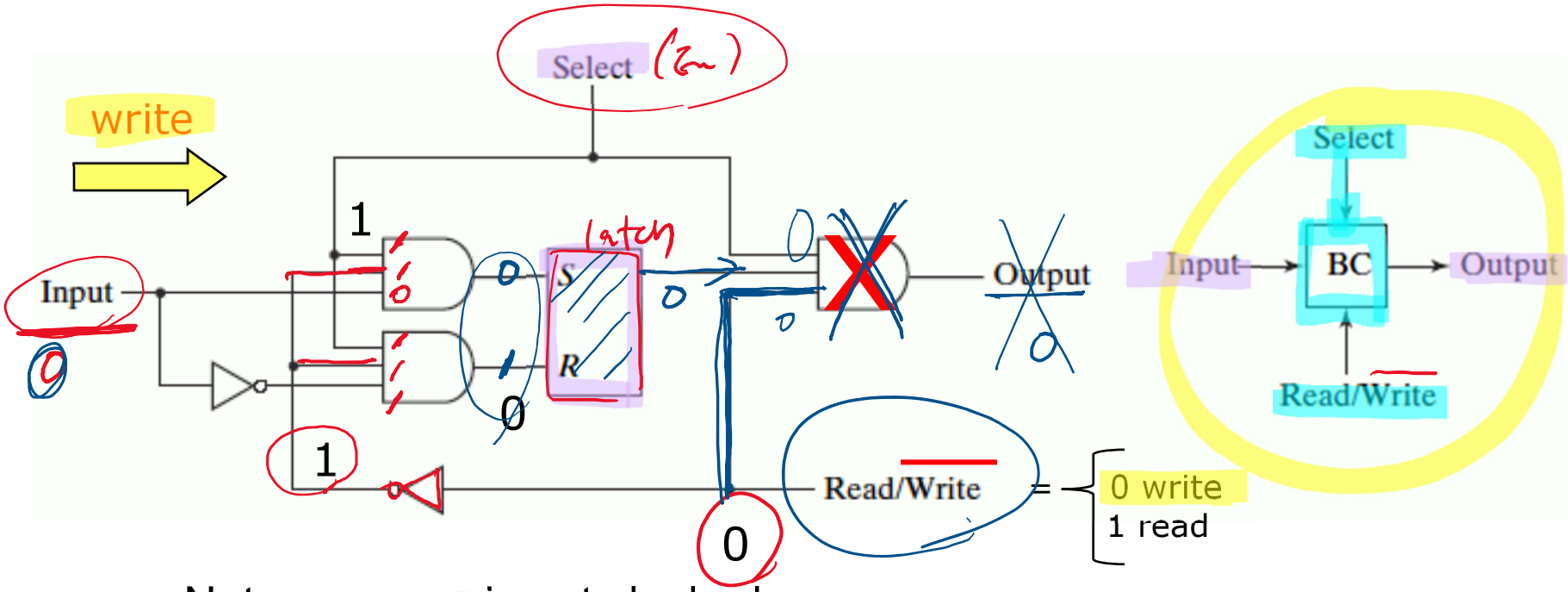
- Control inputs on commercial memories
  - Memory enable
  - Read and write (often *read/$\overline{\text{write}}$*)

Memory enable	Read / $\overline{\text{write}}$	Memory operation
0	X	none
1	0	write to selected word
1	1	read from selected word

- What timing constraints need to be considered?
  - *Access time* = time to read a word
  - *Cycle time* = time to write a word
  - Also: *setup* and *hold time* on address lines

Memory Cell Design

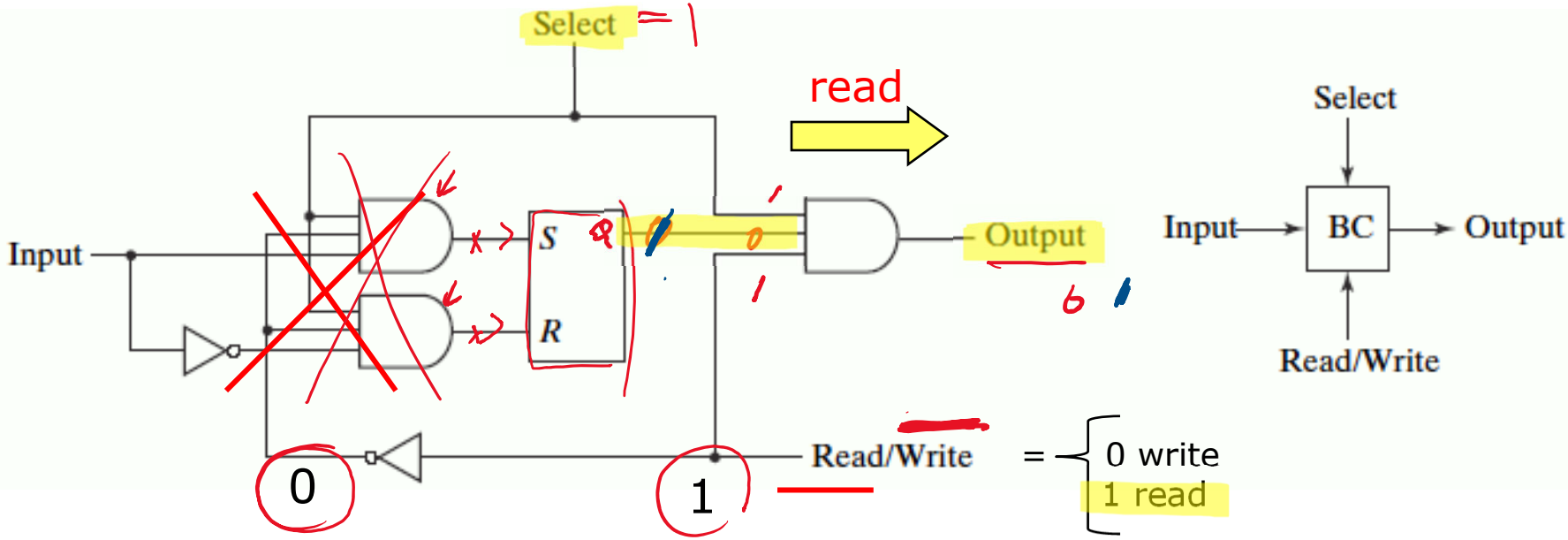
- Binary memory cell implemented with SR latch:



- Note: memory is not clocked

Memory Cell Design

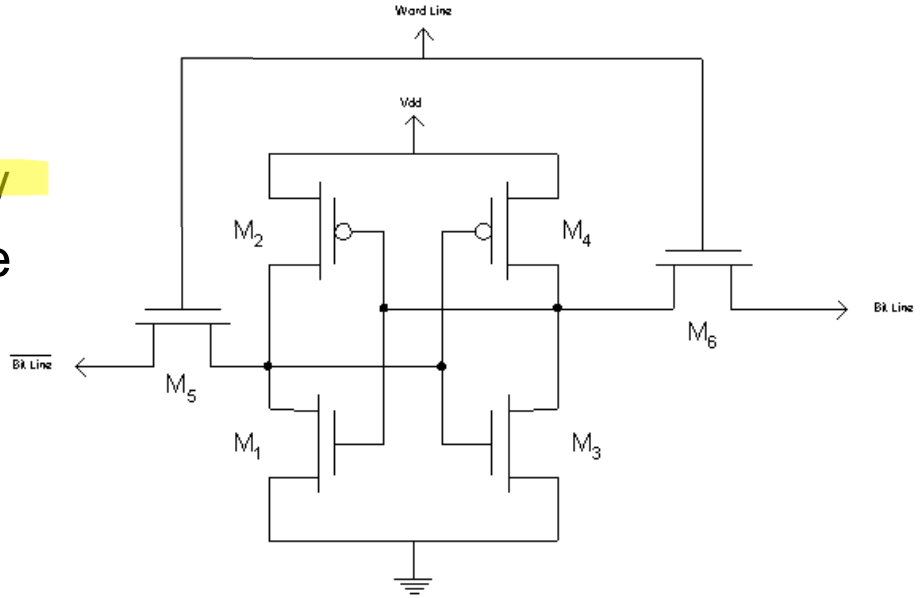
- Binary memory cell implemented with SR latch (nor gate based):



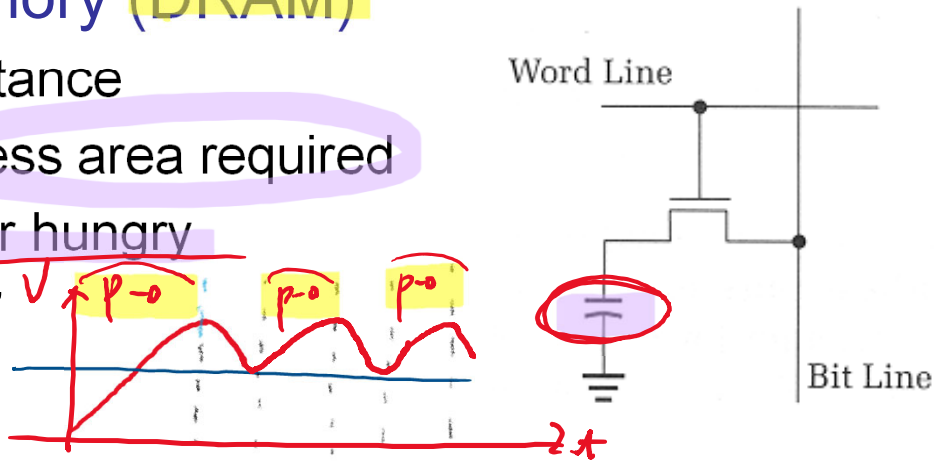
- Note: memory is not clocked

RAM Types

- Static random access memory (SRAM)
  - Operates like a latch
  - Implemented with 6 transistors
  - Fast operation, but power hungry
  - Memory retains information while power is applied

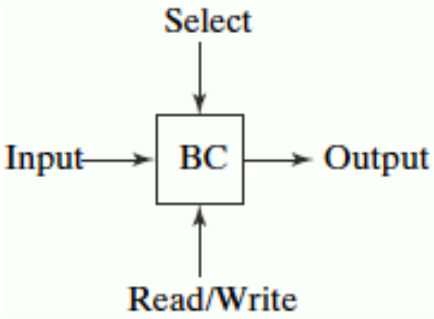


- Dynamic random access memory (DRAM)
  - Data stored as charge in capacitance
  - Implemented with 1 transistor; less area required
  - Slower operation, but less power hungry
  - Memory needs to be "refreshed"

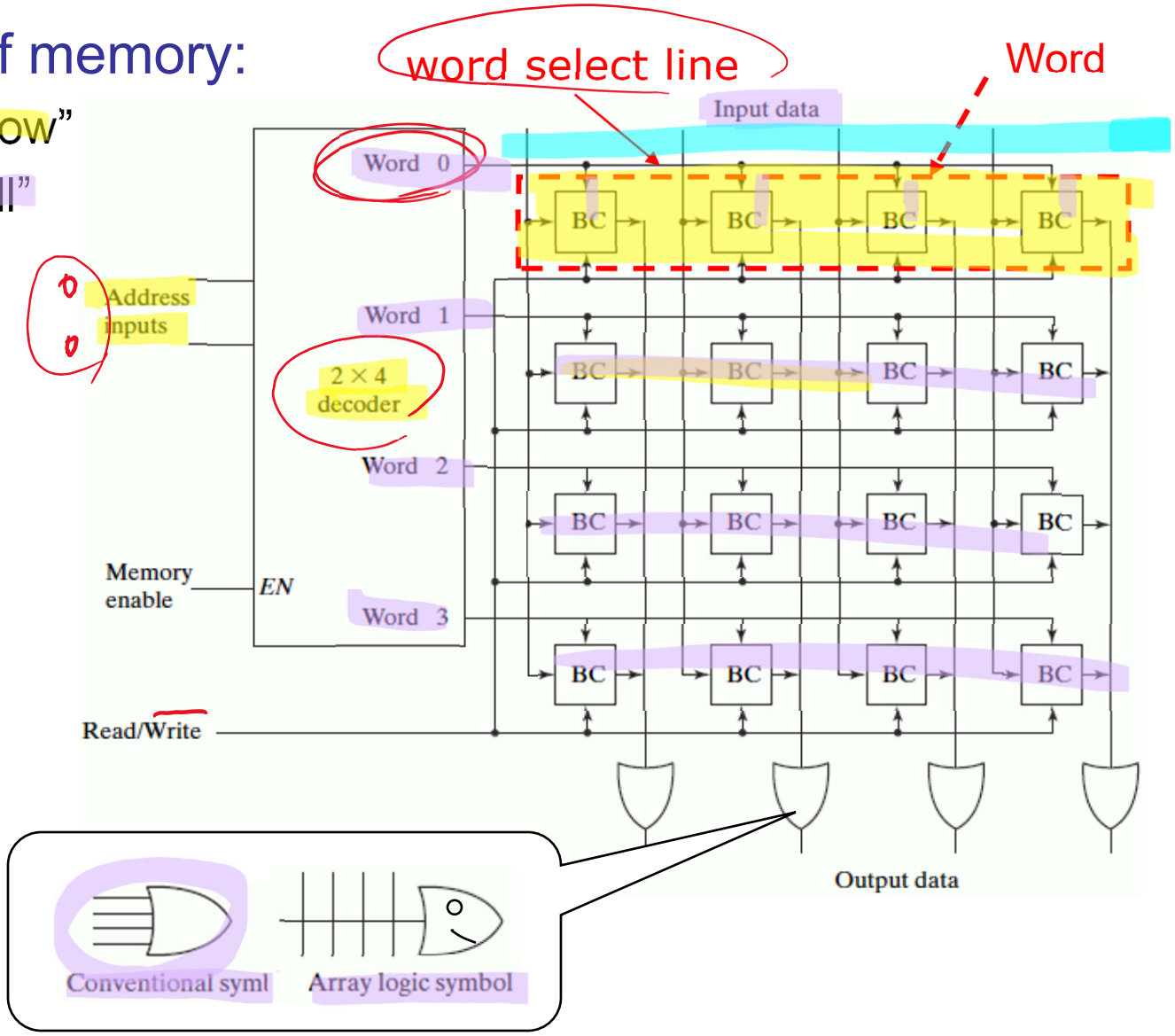


Memory Organization

- Internal layout of memory:
  - One word per "row"
  - Each "binary cell" stores one bit
- Memory cell:

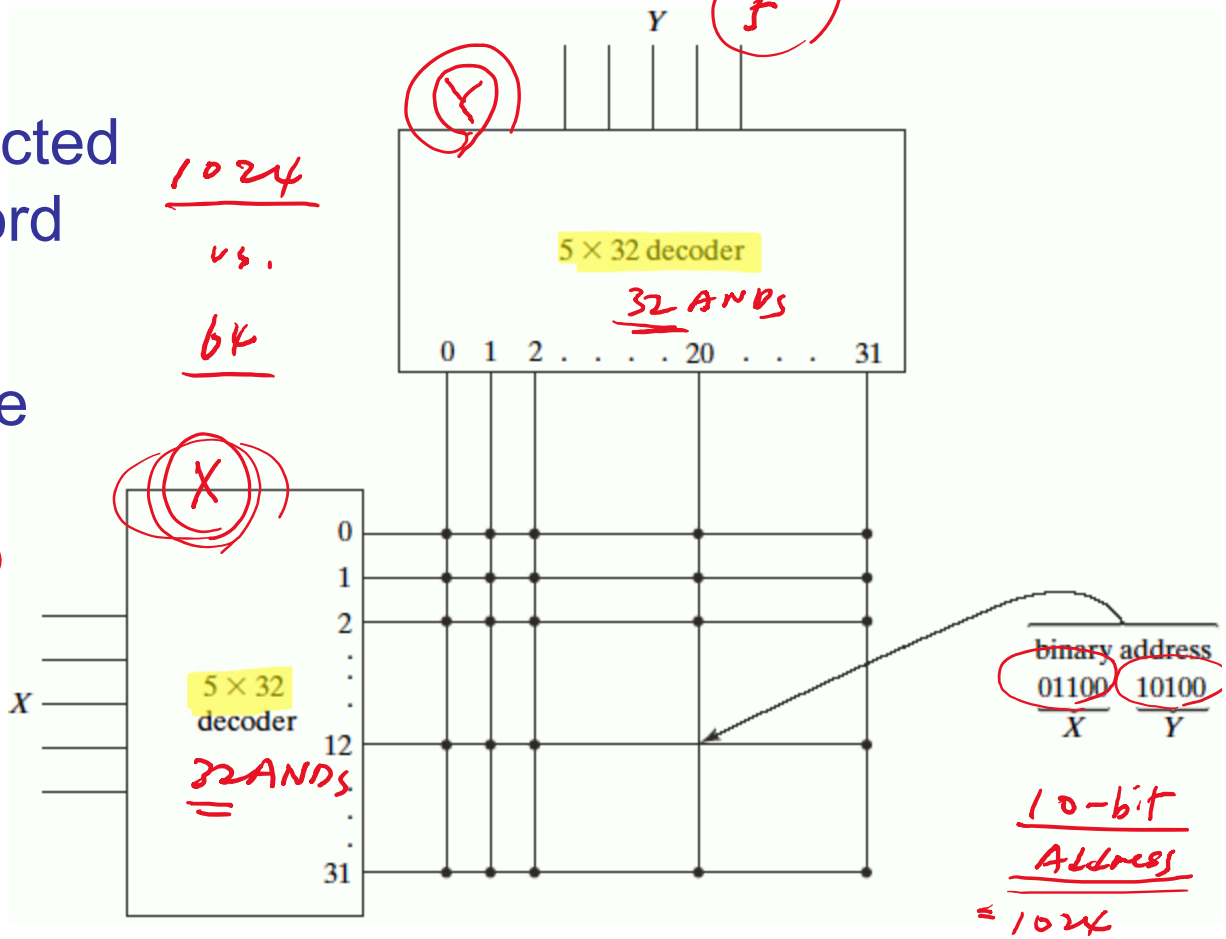


- Decoder:
  - Selects word



Coincident Decoding – Address Splitting

- Split decoders into two dimensions:
  - $k/2 \times 2^{k/2}$ decoder in each dimension
- Coincidence of selected lines determines word
- How many gates are required?
  - 32 + 32 = 64 ANDs with 5 inputs each



Address Multiplexing - DRAM

- Number of pins on memory chip impacts cost
  - How can we reduce the number of pins necessary?
- Address multiplexing
  - Address is transmitted in parts
  - Multiplexing in time
- Example: 64k word DRAM
  - Bit-addressable
    - » 2^{16} address space
  - 16 bits of address are split (multiplexed) in time by strobing
    - » CAS: "Column Address Strobe"
    - » RAS: "Row Address Strobe"