ECE 341

Lecture # 16

Instructor: Zeshan Chishti
zeshan@ece.pdx.edu

November 24, 2014

Portland State University

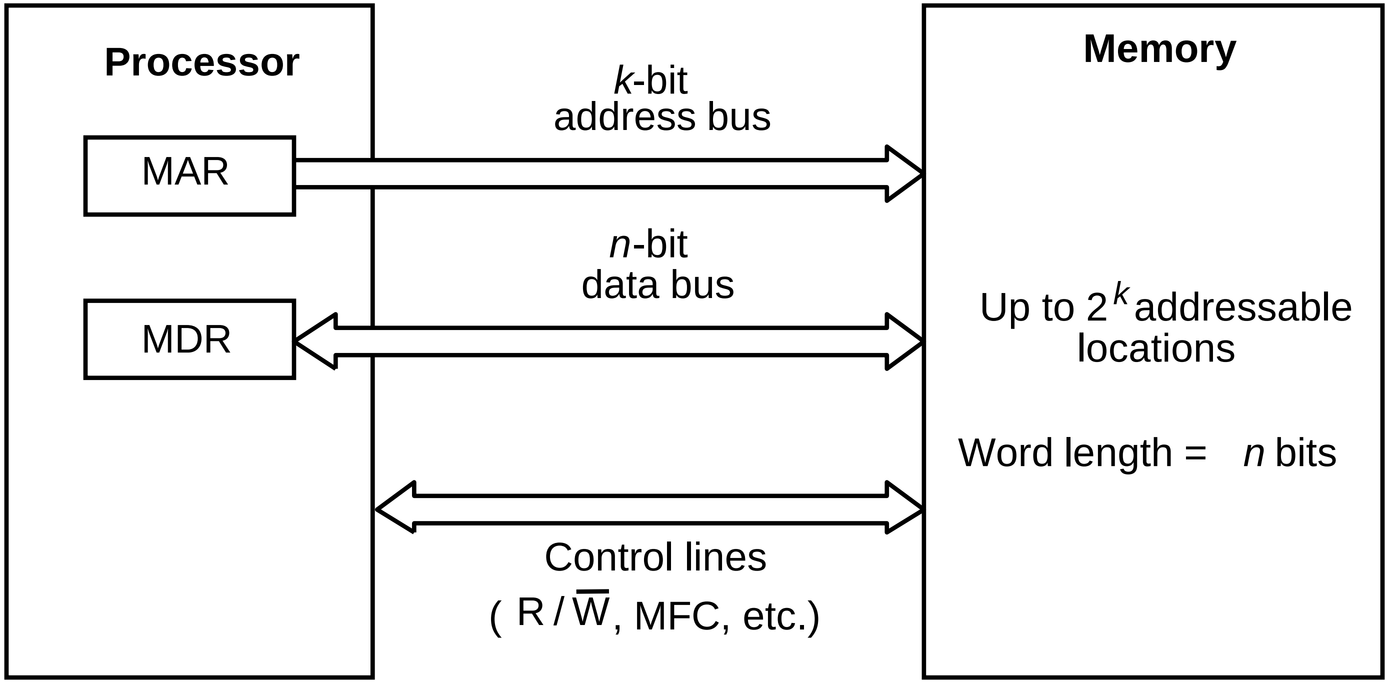
Lecture Topics

- The Memory System
  - Basic Concepts
  - Semiconductor RAM Memories
    - Organization of Memory Devices
    - Static RAM (SRAM)
    - Dynamic RAM (DRAM)
    - Structure of Larger Memories
  - Read-only Memories (ROM)
  - Memory Hierarchy
  - Cache Memories

- <u>Reference:</u>
  - Chapter 8: Sections 8.1, 8.2, 8.3, 8.5 and 8.6 (Pages 268 – 285 and 289 – 290 of textbook)

Basic Concepts

- Each memory location is specified by an *address*
- Max. size of main memory (also called the size of address space) is determined by the no. of address bits
  - e.g., a computer with 32-bit addresses can access 2^{32} = 4 G (giga) memory locations
- Memory transfers usually happen in *word* granularities



Basic Concepts (cont.)

- Measures for memory speed:
  - **Memory access time:** Time elapsed between the initiation of an operation to transfer data from/to memory and the completion of that operation
  - **Memory cycle time:** Time required between initiation of two successive memory accesses

- Most important issue in memory systems design is to provide a computer with as large and fast a memory as possible, within a given cost target

- Cost of a memory device depends on both its capacity (total no. of bits) and its density (bits per unit area)
- For a fixed capacity, higher density => less chip area => less cost per bit
- Different memory technologies have different cost & speed characteristics
- Typically memory speed & memory cost are competing constraints

Random Access Memory (RAM)

- A memory unit is called a *random-access memory* (RAM), if the access time to any location is the same, independent of the location's address

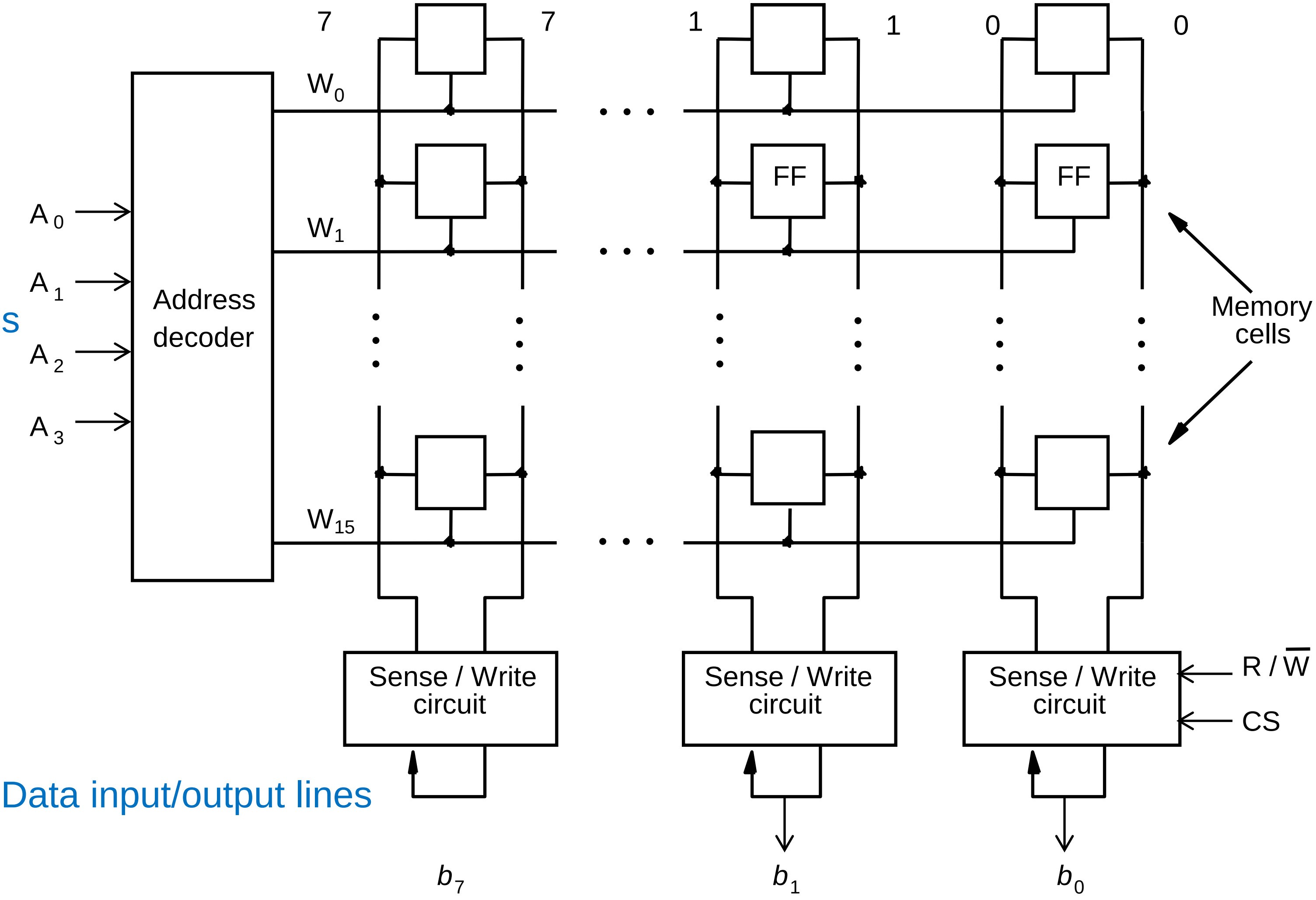
- In non-RAM storage devices, such as magnetic and optical disks, access time depends on the address or position of data

- Two types of RAMs:
  - Static Random Access Memory (SRAM)
  - Dynamic Random Access Memory

Organization of Memory Devices

- A memory chip is made up of a combination of memory cells
- Each memory cell can hold one bit of information
- Memory cells are organized in the form of a 2-dimensional array
- All cells in a row are connected to a common line, known as the "word line"
- Word line is connected to the address decoder
- All cells in a column are connected to a common line, known as the "bit line"
- Sense/write circuits are connected to both the bit lines and the data input/output lines of the memory chip

Example: 16 x 8 memory array



Address lines

Control lines

Data input/output lines

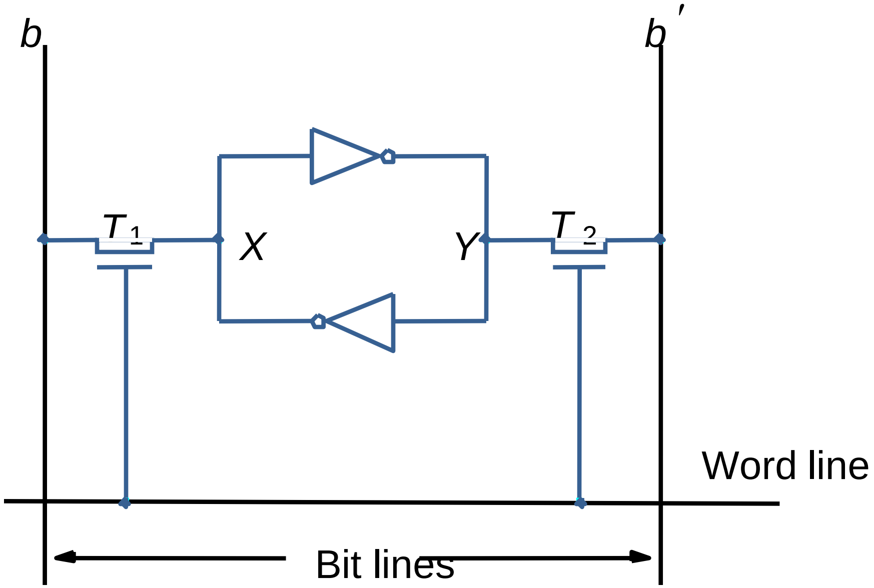
Word size for this memory = 8 bits

Static Random Access Memory (SRAM)

- Consists of circuits that are capable of retaining their state *as long as* the power is applied
- SRAMs are volatile memories, because their contents are lost when power is interrupted
- SRAM use a complex memory cell structure, typically six transistors per memory cell
  - High cost per bit
  - Fast access times (few nanoseconds)
- Used to implement on-chip caches, where speed is of critical concern

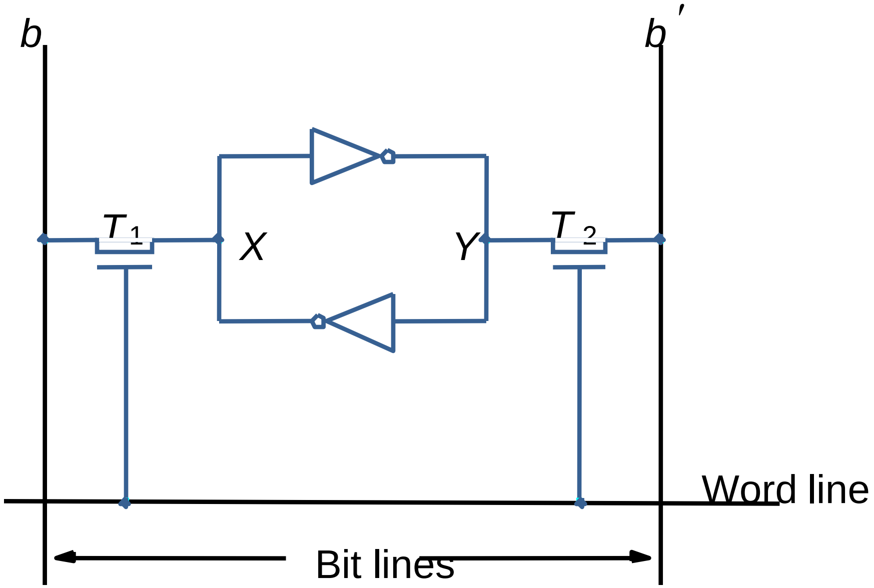
Basic SRAM Cell

- Two inverters cross connected to implement a D flip-flop
- Cell connected to one word line and two bit lines by transistors T1 & T2
- T1 and T2 act as switches.
  - when word line is deactivated, T1 & T2 are turned off and the cell retains state



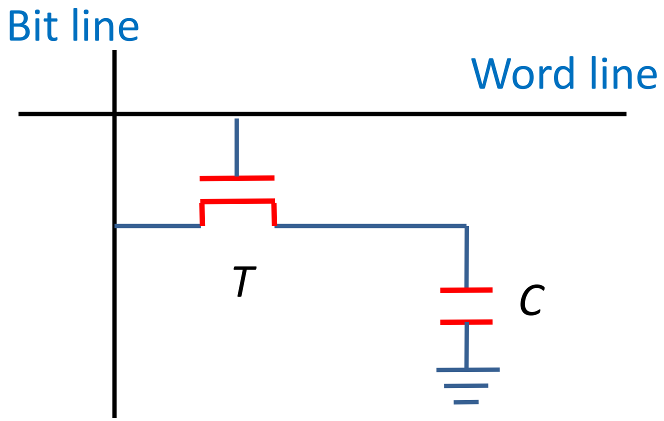
Basic SRAM Cell (cont.)

- **Read operation:**
  - word line is activated to close switches T1 and T2
  - sense/write circuits at the bottom monitor the state of b and b’ to read the bit value
- **Write operation:**
  - word line is activated to close switches T1 and T2
  - sense/write circuits write the bir cell by placing appropriate values on b and b’



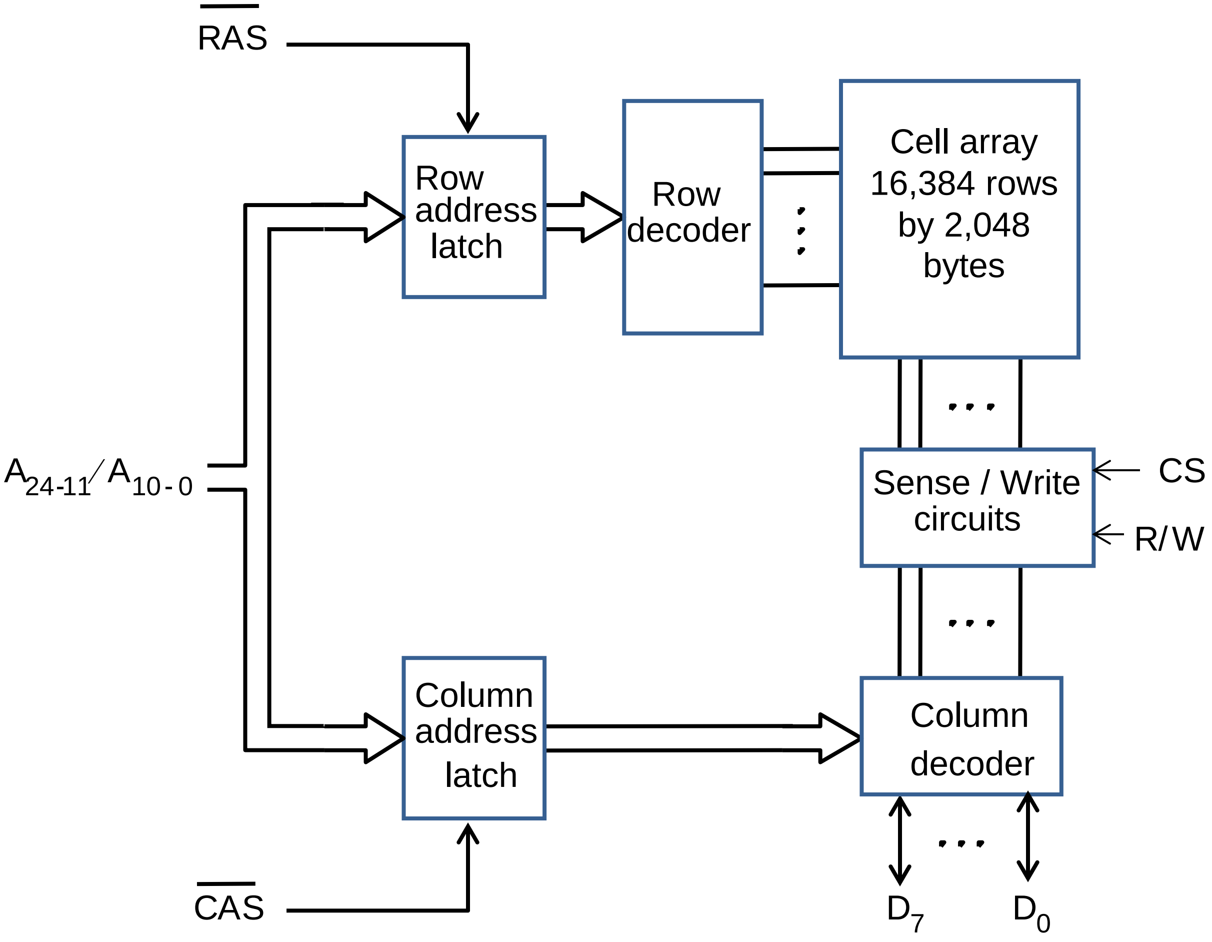
Dynamic Random Access Memory (DRAM)

- Memory cells in DRAMs are much simpler than in SRAMs
  - Each DRAM cell requires only one transistor and one capacitor
- Consequently, DRAMs are less expensive than SRAMs but have longer access times than SRAMs
- DRAMs are typically used as main memory, not as cache
- DRAM cells lose their state over time and need to be periodically *refreshed* to prevent loss of stored state



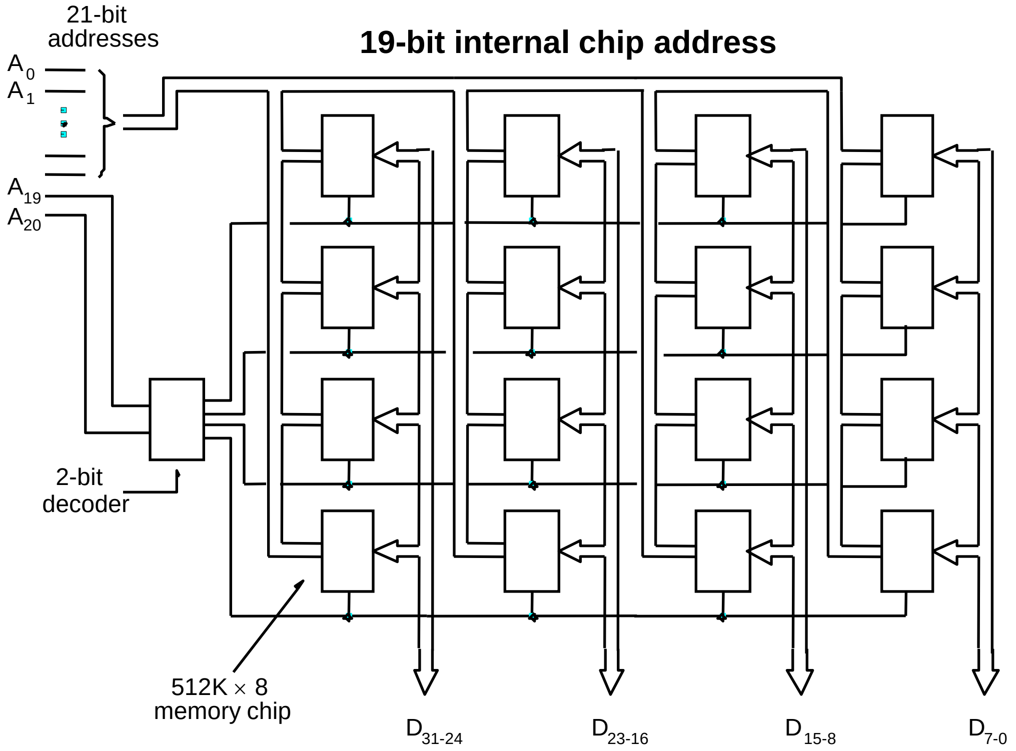
Capacitor discharges slowly to lose the stored state over time

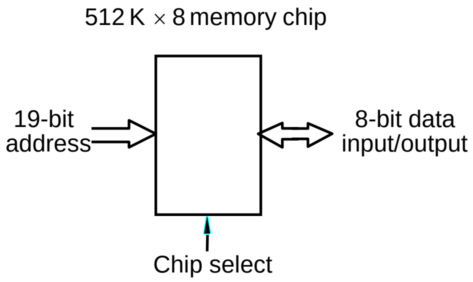
Example: A 32M x 8 DRAM Chip



- Word size = 8-bits (1B), so there are 8 data lines ($D_7 – D_0$)
- There are 16384 rows, each row can store 2048 bytes (16348 * 2048 = 32M words)
- 14 bits ($A_{24} – A_{11}$) to select a row, and 11 bits ($A_{10} – A_0$) to select a byte within a row
- First apply the row address, RAS signal latches the row address. Then apply the column address, CAS signal latches the column address.
- Memory timing is controlled by a specialized unit which generates RAS and CAS
- This DRAM is asynchronous

Example: Large Memories





Problem: Implement a memory unit of 2M words of 32 bits each by using 512Kx8 memory chips.

Solution: (shown on left)

- We need 16 total chips, arranged in 4 rows and 4 columns
- Each chip provides 8 bits of the 32-bit word
- A chip is selected by setting its chip select control line to 1.
- On a memory access, the 2-bit decoder decides which row provides the required word, all chips in that row are selected

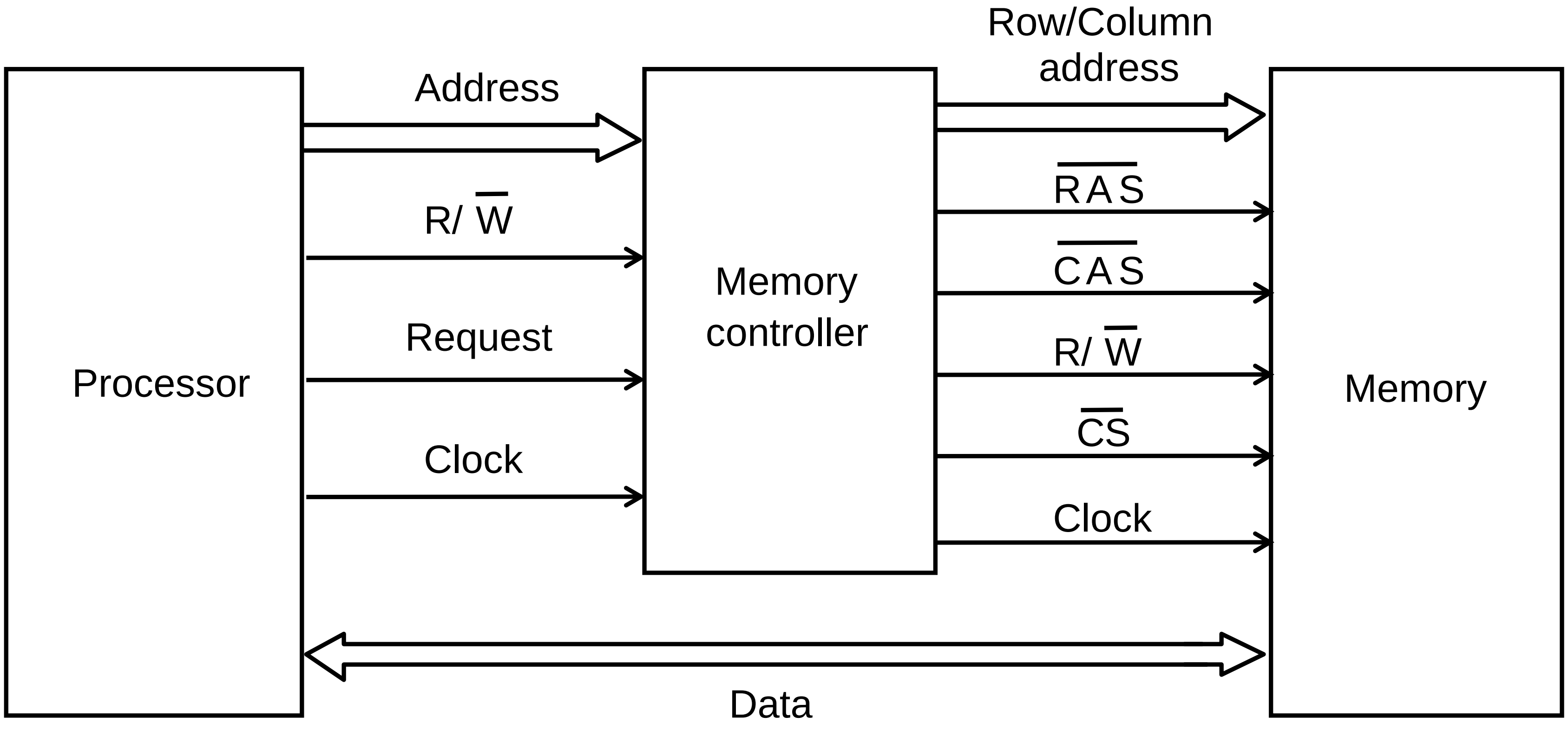
Fast Page Mode

- Suppose we want to read the consecutive bytes in a row
- This can be done without having to re-read the entire row every time
  - Add a latch (called row buffer) at the output of sense/write circuits
  - When the row is selected, the sense circuit reads the row contents and transfers them into the row buffer
  - Subsequent accesses can read the row contents from the row buffer
- Consecutive sequence of column addresses can be applied under the control signal CAS, without reselecting the row
  - Allows a data block to be transferred much faster than random accesses
- This capability is referred to as the fast page mode feature.

Memory Controller

- Recall that accessing a DRAM chip requires multiplexed addresses,
- Each DRAM address is divided into two parts:
  - High-order address bits select a row in the DRAM array. They are provided first, and latched using RAS signal
  - Low-order address bits select a column in the row. They are provided later, and latched using CAS signal
- However, the address sent by processor includes all the address bits at the same time
- In order to achieve the multiplexing, memory controller circuit is inserted between the processor and memory
- Memory controller acts as a bridge between processor and memory
- It receives a memory request from the processor and then generates the appropriate signals (R/W, row and column address, RAS, CAS) with appropriate timing to access the memory

Memory Controller (cont.)



Read-only Memories (ROM)

- SRAM and DRAM chips are volatile
  - Retain contents only while power is turned on
- Many applications require memory devices that retain their contents after power is turned off (non-volatile memories)
  - For example, the program that starts the bootstrap process of loading the OS from hard disk into main memory
- Non-volatile memory is read in same manner as volatile memory
- A special writing process is needed to place information into a non-volatile memory
- Since normal operation involves only reading the stored data, this type of memory is called a *Read-Only memory*

Example: Flash Memory

- Flash memory
  - Reads can happen at single cell granularity, but writes must happen at ***block*** granularity (a block is several thousand bits)
  - Before writing to a location, the entire block must be erased
    - Write usually much slower (10x to 100x) than reads
  - Flash devices have a limited endurance; cells *wear out* after a certain number of writes
  - Flash devices have higher density than DRAM => Lower cost/bit
  - Power consumption of flash memory is very low, making it attractive for use in battery-powered devices
  - Single flash chips are not sufficiently large, so larger memory modules are implemented using flash cards and flash drives

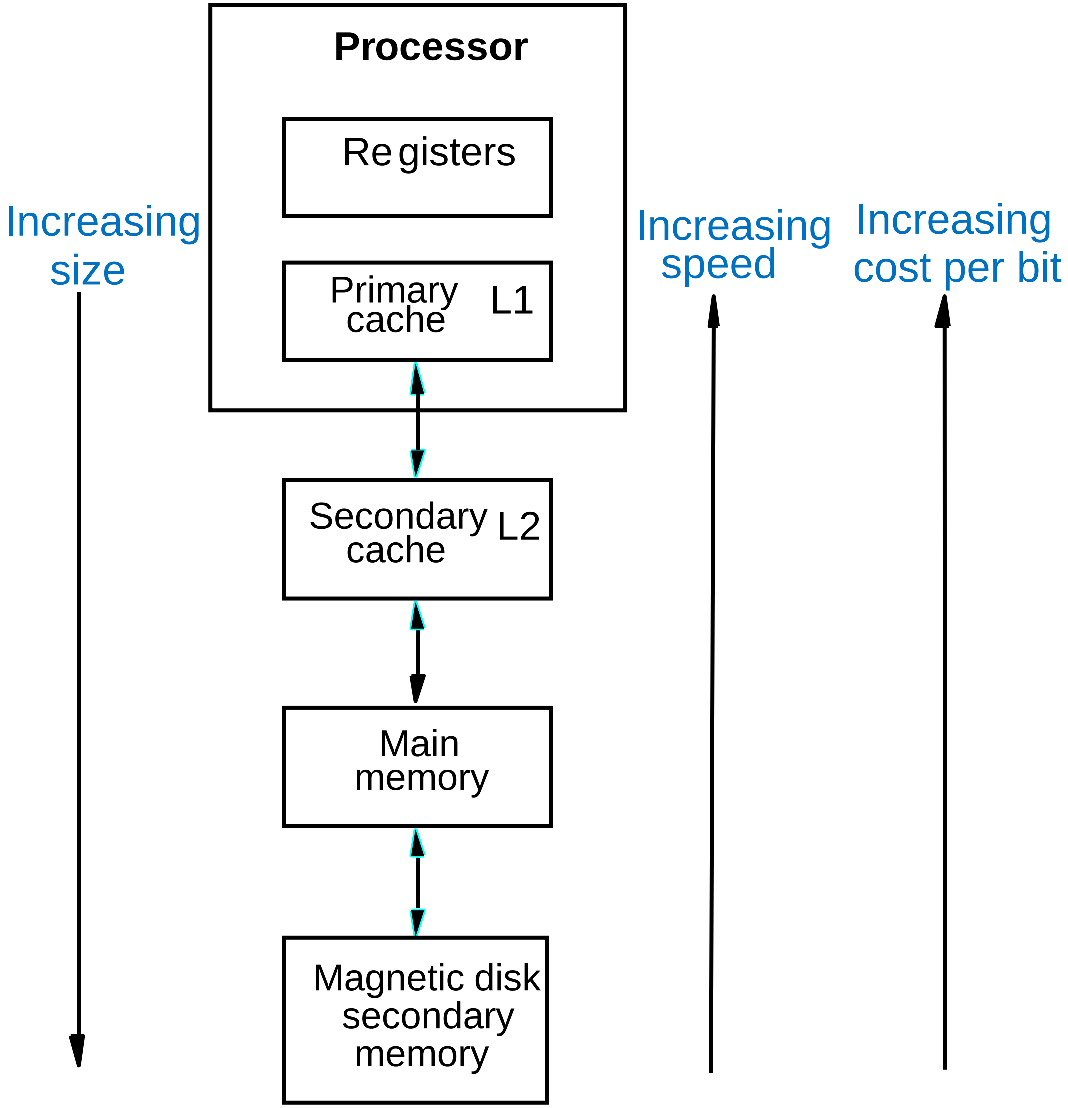
Memory System Design Tradeoffs

- A big challenge in memory system design is to provide a sufficiently large memory capacity, with reasonable speed at an affordable cost

- **SRAM**
  - Complex basic cell circuit => fast access, but high cost per bit
- **DRAM**
  - Simpler basic cell circuit => less cost per bit, but slower than SRAMs
- **Flash memory and Magnetic disks**
  - DRAMs provide more storage than SRAM but less than what is necessary
  - Disks provide a large amount of storage, but are much slower than DRAMs

No single memory technology can provide both large capacity and fast speed at an affordable cost

Memory Hierarchy



- Memory system implemented as a **multi-level hierarchy** to provide both **large capacity** and **fast access**
- Higher levels made of small but fast SRAMs
    - ➢ Fast access speed in the common case
- Lower levels made of DRAM and hard disk
    - ➢ Large capacity at low cost

Key idea is to **reduce average memory access time** by bringing instructions and data that will be used in the near future as close to the processor as possible

Cache Memories

- **Memory Wall Problem:**
  - Processor is much faster than the main memory
  - Main memory speed cannot be increased beyond a certain point
  - Processor has to spend much of its time waiting while instructions/data are being fetched from memory

- **Solution: Cache Memory**
  - Cache memory is an architectural arrangement that makes the main memory appear faster to the processor than it really is
  - Cache memory is based on the property of computer programs that is known as "locality of reference"

Locality of Reference

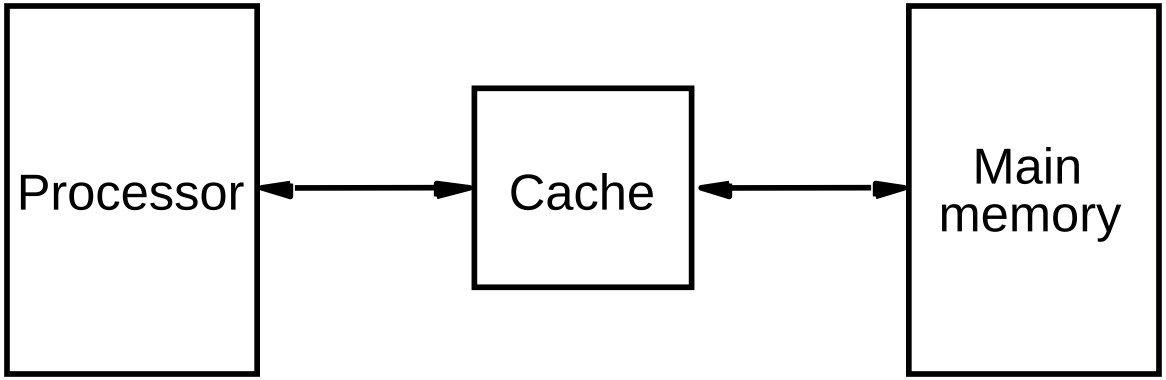
- Analysis of programs indicates that many instructions in localized areas of a program are executed repeatedly during some period of time, while other instructions are executed relatively less frequently
  - Example: loops, nested loops or few procedures calling each other repeatedly
  - This is called "locality of reference"

- Temporal locality of reference
  - Recently executed instruction is likely to be executed again very soon
  - Recently accessed data is likely to be accessed very soon

- Spatial locality of reference
  - Instructions/data with addresses close to a recently accessed instruction/data are likely to be accessed soon

A cache is designed to take advantage of both types of "locality of reference"

Use of a Cache Memory



- Small and fast SRAM cache inserted between processor and main memory
- Data organized at the granularity of cache blocks
- When the processor issues a request for a memory address, an entire block (e.g., 64 bytes) is transferred from the main memory to the cache
- Later references to same address can be serviced by the cache (temporal locality)
- References to other addresses in this block can also be serviced by the cache (spatial locality)

Higher locality => More requests serviced by the cache