Santa Clara University

COEN 210

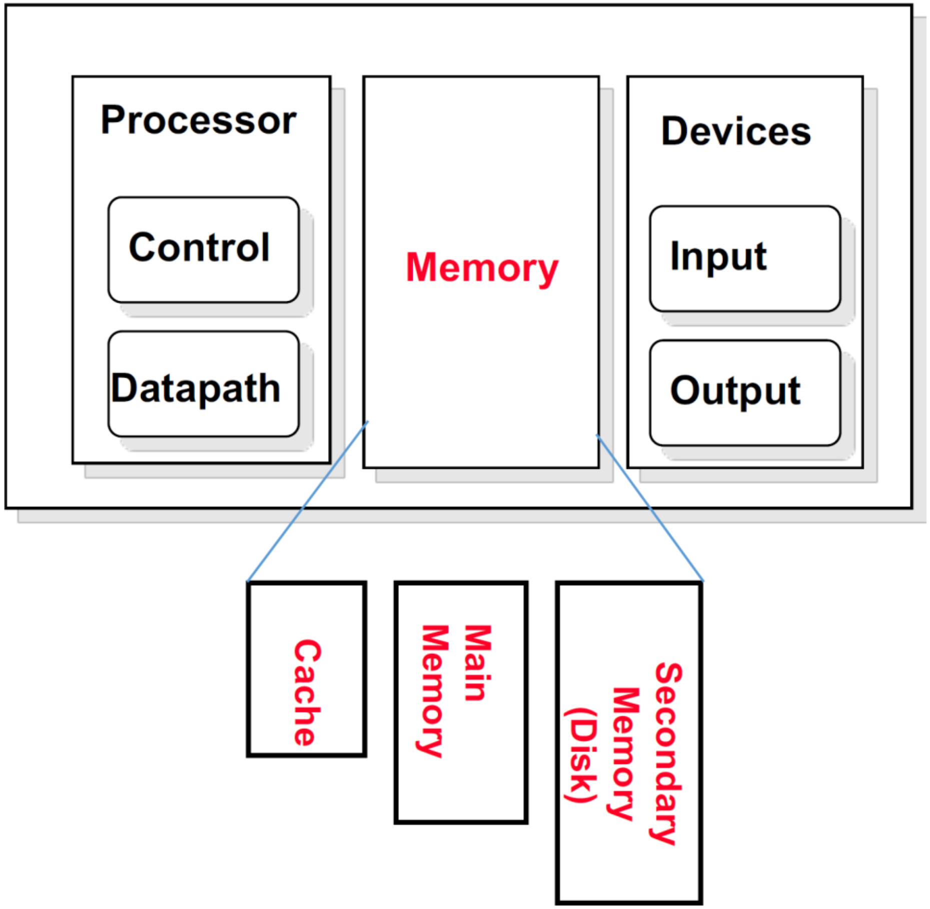
Chapter 5 – Memory Hierarchy

Yan Cui
ycui4@scu.edu

Content

- chapter 5 - memory hierarchy
  - Chapter 5.1 – Memory Technology and Organization
  - Chapter 5.2 - Caches
  - Chapter 5.3 - Virtual Memory

Review- Major Components of a Computer



Memory System Revisited

- Maximum size of memory is determined by addressing scheme

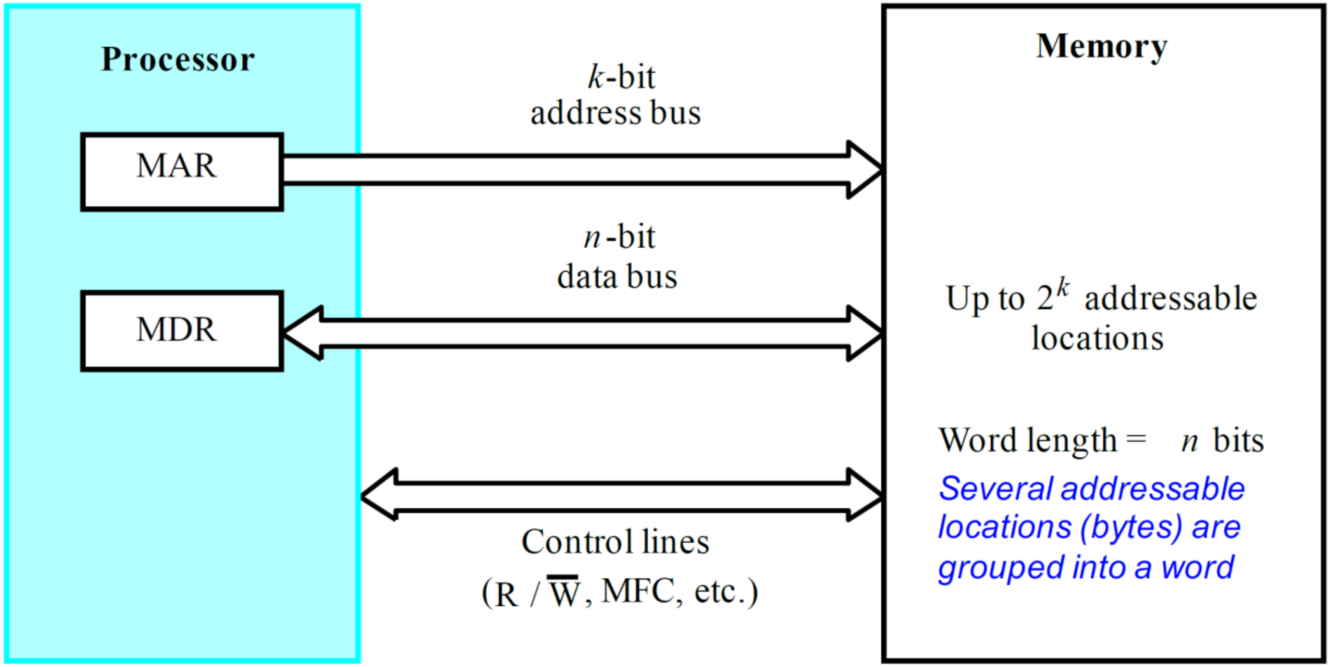
E.g.

16-bit addresses can only address $2^{16} = 65536$ memory locations

- Most machines are byte-addressable
- each memory address location refers to a byte
- Most machines retrieve/store data in words
- Common abbreviations
  - $1\text{k} \approx 2^{10}$ (kilo)
  - $1\text{M} \approx 2^{20}$ (Mega)
  - $1\text{G} \approx 2^{30}$ (Giga)
  - $1\text{T} \approx 2^{40}$ (Tera)

Simplified View

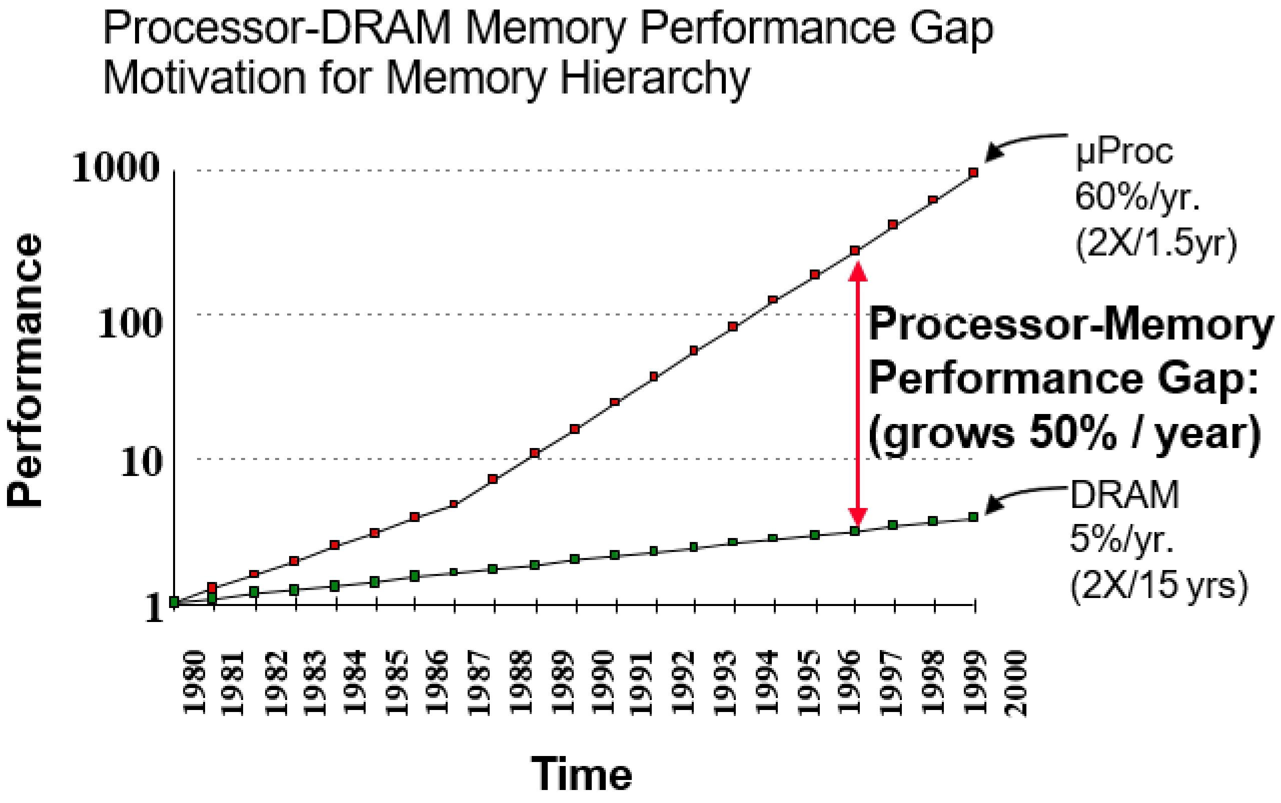
- Data transfer takes place through
- MAR: memory address register
- MDR: memory data register



Big Picture

- Processor usually runs much faster than main memory:
  - Small memories are fast, large memories are slow.
  - Use a cache memory to store data in the processor that is likely to be used.
- Main memory is limited:
  - Use virtual memory to increase the apparent size of physical memory by moving unused sections of memory to disk (automatically).
  - A translation between virtual and physical addresses is done by a memory management unit (MMU)
  - To be discussed in later lectures

Who Cares About the Memory Hierarchy?



Processor-DRAM Memory Performance Gap
Motivation for Memory Hierarchy

Impact on Performance

- Suppose a processor executes at
  - Clock Rate = 200 MHz (5 ns per cycle)
  - CPI = 1.1
  - 50% arith/logic, 30% ld/st, 20% control
- Suppose that 10% of memory operations get 50 cycle miss penalty
- CPI = ideal CPI + average stalls per instruction
  = 1.1(cyc) +(0.30 (datamops/ins)
  x 0.10 (miss/datamop) x 50 (cycle/miss))
  = 1.1 cycle + 1.5 cycle
  = 2.6
- 58 % of the time the processor is stalled waiting for memory!
- a 1% instruction miss rate would add an additional 0.5 cycles to the CPI!

Need for Large Memory

- Small memories are fast
- So just write small programs

  “640 K of memory should be enough for anybody”

  -- Bill Gates, 1981
- Today’s programs require large memories
  - Powerpoint 2003 – 25 megabytes
  - Data base applications may require Gigabytes of memory
  - Big data may require Terabytes, even Yottabytes

The Goal: Illusion of large, fast, cheap memory

- How do we create a memory that is large, cheap and fast (most of the time)?
  - Hierarchy
  - Parallelism

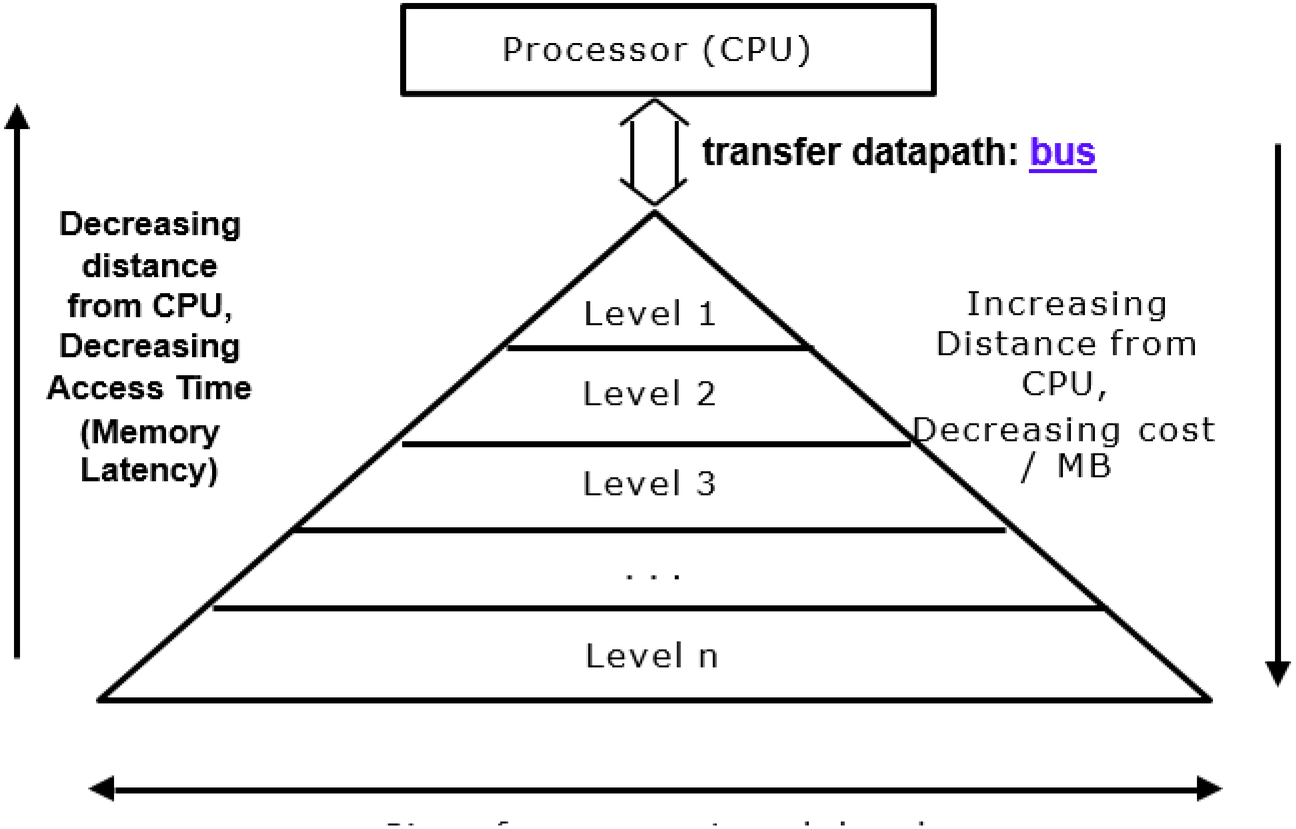
- Strategy: Provide a Small, Fast Memory which holds a subset of the main memory – called cache
  - Keep frequently-accessed locations in fast cache
  - Cache retrieves more than one word at a time
  - Sequential accesses are faster after first access

Memory Hierarchy

- Hierarchy of Levels
  - Uses smaller and faster memory technologies close to the processor
  - Fast access time in highest level of hierarchy
  - Cheap, slow memory furthest from processor

- The aim of memory hierarchy design is to have access time close to the highest level and size equal to the lowest level

Memory Hierarchy Pyramid


Processor (CPU)
transfer datapath: bus
Decreasing distance from CPU, Decreasing Access Time (Memory Latency)
Level 1
Level 2
Level 3
. . .
Level n
Increasing Distance from CPU, Decreasing cost / MB

The structure of memory hierarchy

Why Hierarchy works: Natural Locality

- The principle of locality:
  - Programs access a relatively small portion of the address space at any second

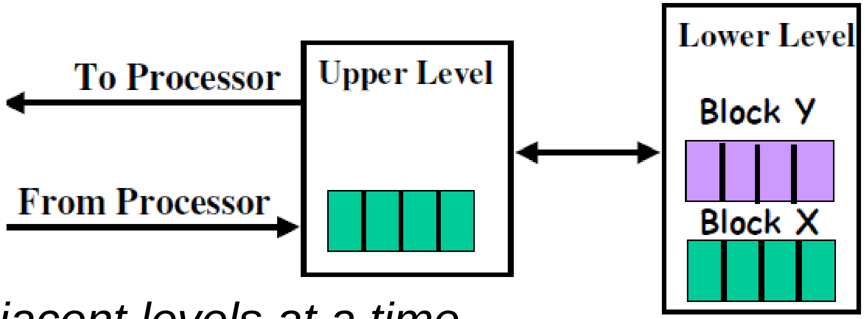
- Temporal Locality (Locality in Time): Recently accessed data tend to be referenced again soon
  - E.g., instructions in a loop, induction variables
  - Keep most recently accessed data items closer to the processor
- Spatial Locality (Locality in Space): if an item is referenced, nearby items will tend to be referenced soon
  - Libraries put books on the same topic together on the same shelves to increase spatial locality
  - E.g., sequential instruction access, array data
  - Move blocks consisting of contiguous words closer to the processor

Memory Hierarchy

- A memory hierarchy consists of multiple levels of memory with different speed and size
  - The faster memories are more expensive per bit than slower memories and
  - Thus, are smaller
  - Closer to the processor
- Data are copied between only two adjacent levels at a time

Terminology

- Block / line: the minimum unit of information that is present (or not) in a cache
- Hit: data appears in upper level in block X
- Hit Rate: the fraction of memory accesses found in the upper level
- Hit Time: Time to access the upper level: Time to access the block + Time to determine hit/miss
- Miss: data needs to be retrieved from a block in the lower level (Block Y)
- Miss Rate = 1 - (Hit Rate)
- Miss Penalty: Time to replace a block in the upper level +Time to deliver the block to the processor
- Note: Hit Time << Miss Penalty

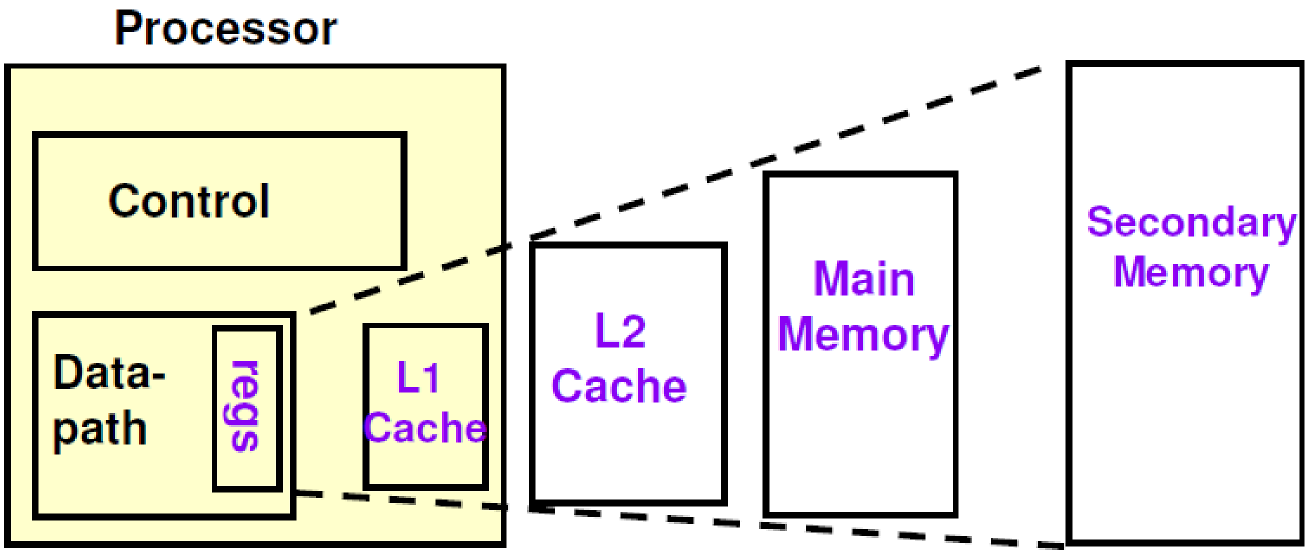


Data are copied between only two adjacent levels at a time

Bandwidth v.s. Latency

- Bandwidth:
  - talking about the “number of bits/bytes per second” when transferring a block of data steadily.
- Latency:
  - amount of time to transfer the first word of a block after issuing the access signal.
  - Usually measure in “number of clock cycles” or in ns/us.

Memory Hierarchy of a Modern Computer System

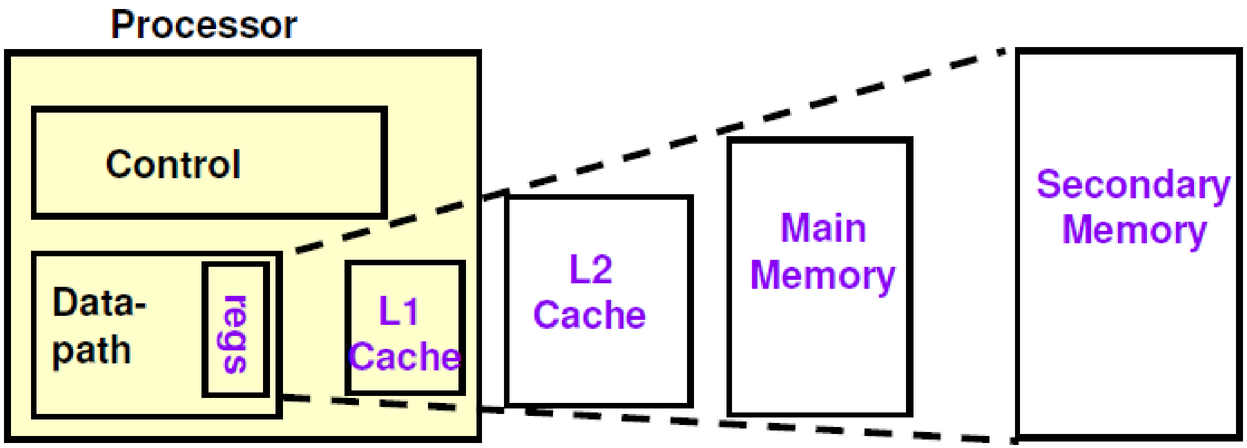


Speed(ns):	1ns	2ns	6ns	100ns	10,000,000ns
Size (MB):	0.0005	0.1	1-4	1000-6000	500,000
Cost ($/MB):	--	$10	$3	$0.01	$0.002
Technology:	Regs	SRAM	SRAM	DRAM	Disk

- Cache - Main memory: Speed
- Main memory – Disk (virtual memory): Capacity

How is the hierarchy managed?

- Registers « Cache
  - By the compiler (or assembly language Programmer)
- Cache « Main Memory
  - By hardware
- Main Memory « Disks
  - By combination of hardware and the operating system
  - virtual memory
  - Also by the programmer (in case of files)



Memory Background

- Performance of Main Memory:
  - Latency: Cache Miss Penalty
    - *Access Time*: time between request and word arrives
    - *Cycle Time*: time between requests
  - Bandwidth: I/O & Large Block Miss Penalty (L2)
- Main Memory is *DRAM*: Dynamic Random Access Memory
  - Dynamic since needs to be refreshed periodically (8 ms)
  - Addresses divided into 2 parts (Memory as a 2D matrix):
    - *RAS* or *Row Access Strobe*
    - *CAS* or *Column Access Strobe*
- Cache uses *SRAM*: Static Random Access Memory
  - No refresh (6 transistors/bit vs. 1 transistor/bit)
  - Address not divided

Memory Technology

Four primary technologies used today in memory hierarchies

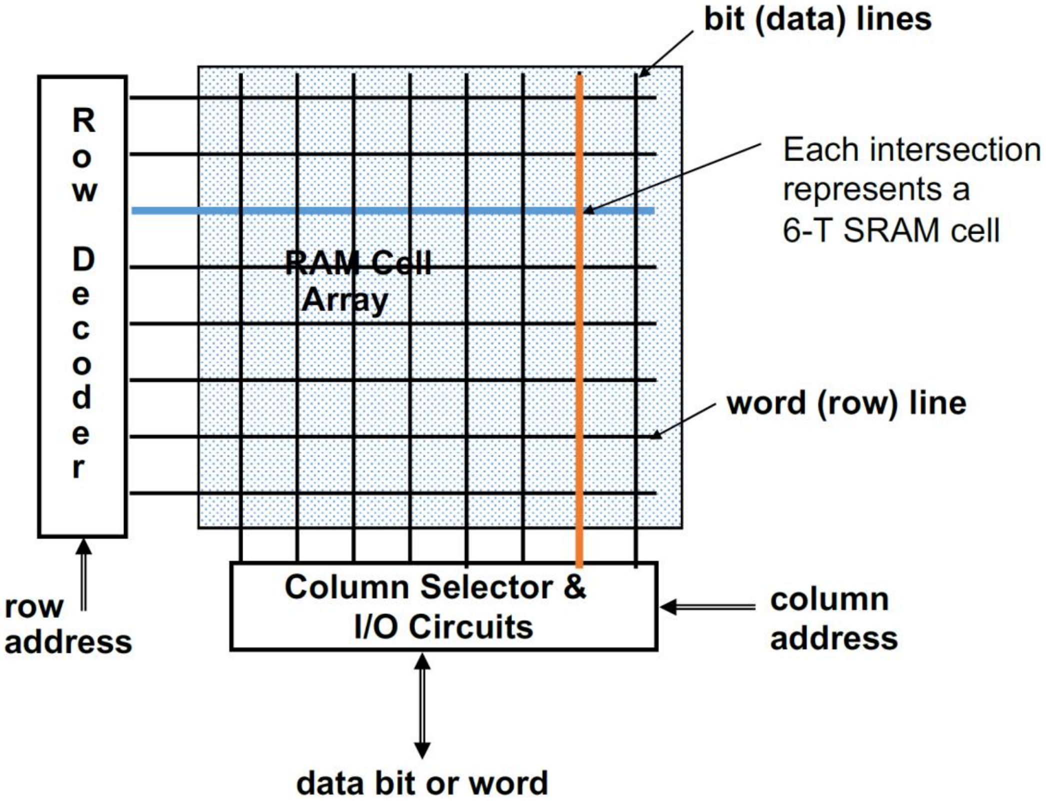
- Static RAM (SRAM)
  - 0.5ns – 2.5ns, $500 – $1000 per GB
- Dynamic RAM (DRAM)
  - 50ns – 70ns, $3 – $6 per GB
- Magnetic disk
  - 5ms – 20ms, $0.01– $0.02 per GB
- Flash
  - 5,000ns – 50,000ns, $0.06 - $0.12 per GiB

- Ideal memory
  - Access time of SRAM
  - Capacity and cost/GB of disk

SRAM Technology

- Integrated Circuits that are memory arrays with a single access port
- Provide either a read or a write
- Fixed access time to any data
- the read and write access times may differ
- Access time is close to the cycle time
- Use six to eight transistors per bit

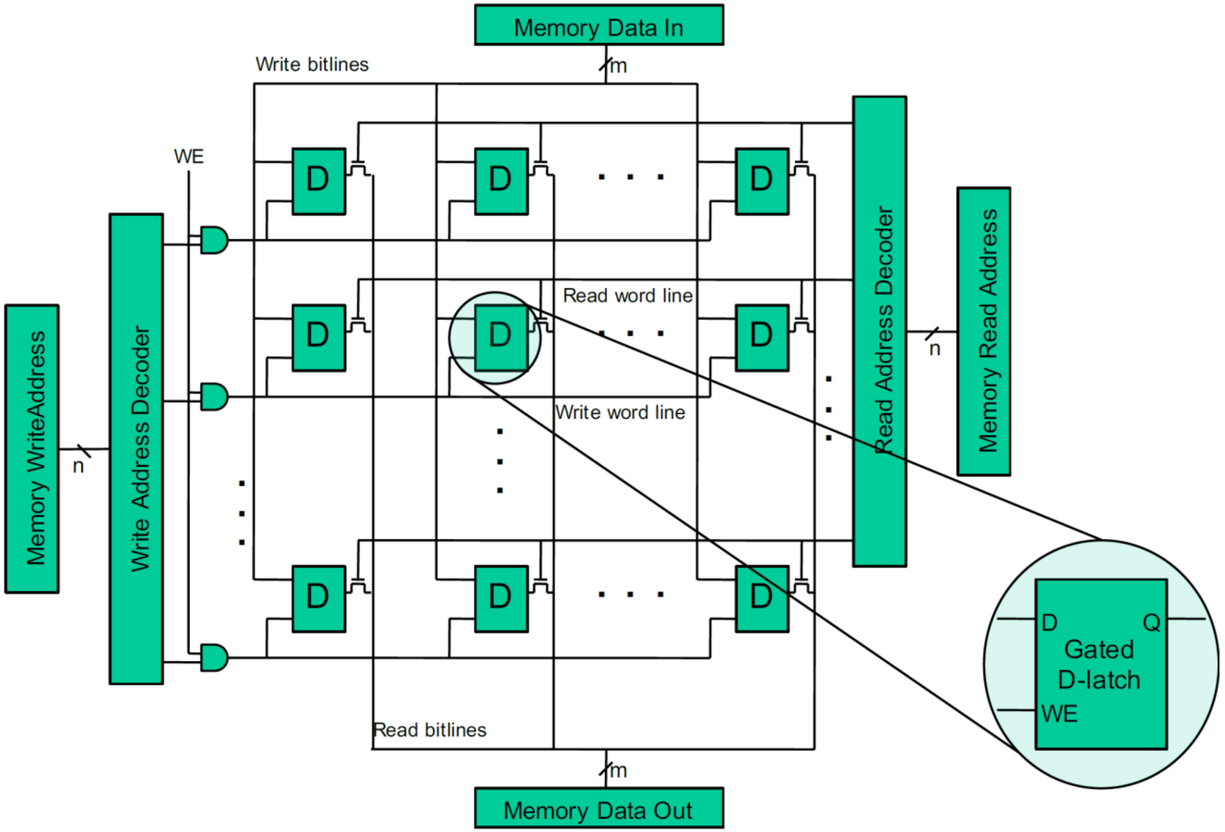
Classical SRAM Organization



- One memory row holds a block of data, so the column address selects the requested bit or word from that block

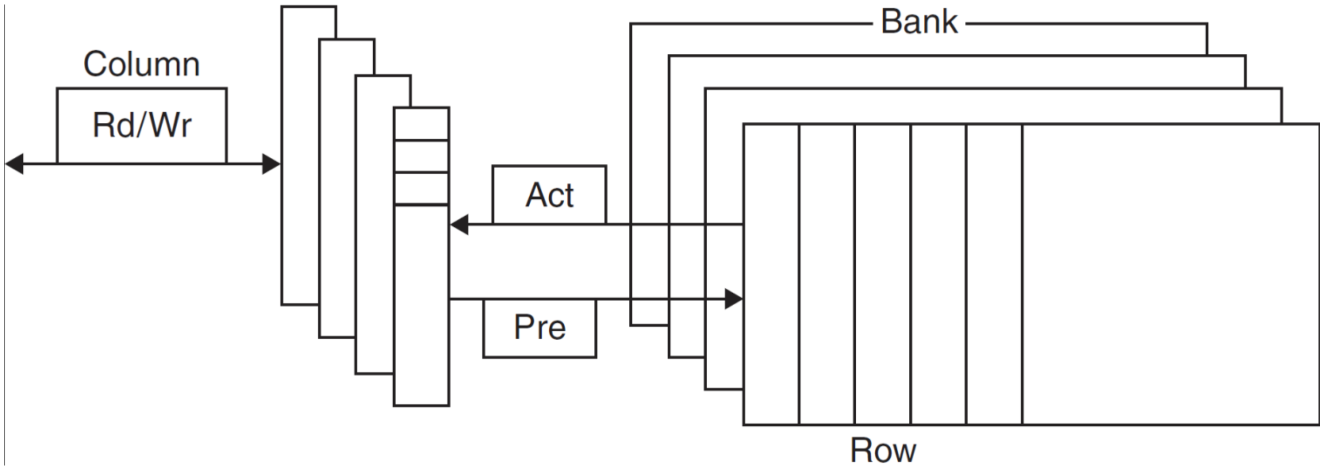
Classical SRAM Organization

- Latch based memory



DRAM Technology

- Data stored as a charge in a capacitor
  - Single transistor used to access the charge
  - Must periodically be refreshed
    - Read contents and write back
    - Performed on a DRAM "row"



Internal organization of a DRAM

DRAM Generations

Year	Capacity	$/GB
1980	64Kbit	$6480,000
1983	256Kbit	$1980,000
1985	1Mbit	$720,000
1989	4Mbit	$128,000
1992	16Mbit	$30,000
1996	64Mbit	$9000
1998	128Mbit	$900
2000	256Mbit	$840
2004	512Mbit	$150
2007	1Gbit	$40
2018	16 Gbit	$6

Advanced DRAM Organization

- Bits in a DRAM are organized as a rectangular array
  - DRAM accesses an entire row
  - Burst mode: supply successive words from a row with reduced latency
- Double data rate (DDR) DRAM
  - Transfer on rising and falling clock edges
  - Twice as much bandwidth
- Quad data rate (QDR) DRAM
  - Separate DDR inputs and outputs

Synchronous DRAM (SDRAM)

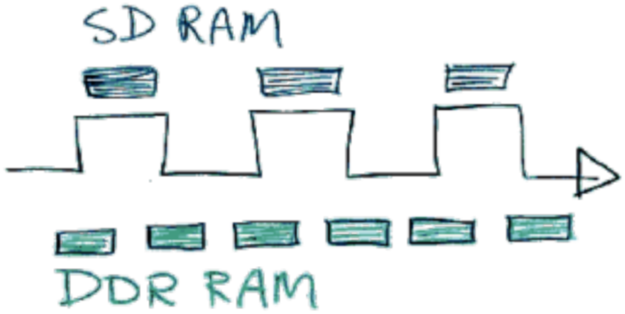
- The common type used today as it uses a clock to synchronize the operation.
- The refresh operation becomes transparent to the users.
- All control signals needed are generated inside the chip.
- The initial commercial SDRAM in the1990s were designed for clock speed of up to 133MHz.
- Today’s SDRAM chips operate with clock speeds exceeding 1 GHz.

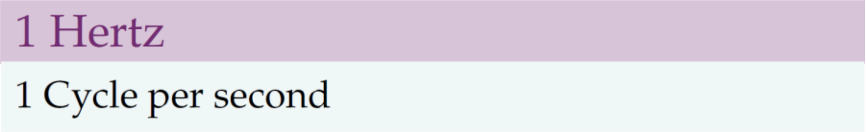
Memory modules are used to hold several SDRAM chips
And are the standard type used in a computer’s
motherboard, of size like 4GB or more.

Double Data Rate (DDR) SDRAM

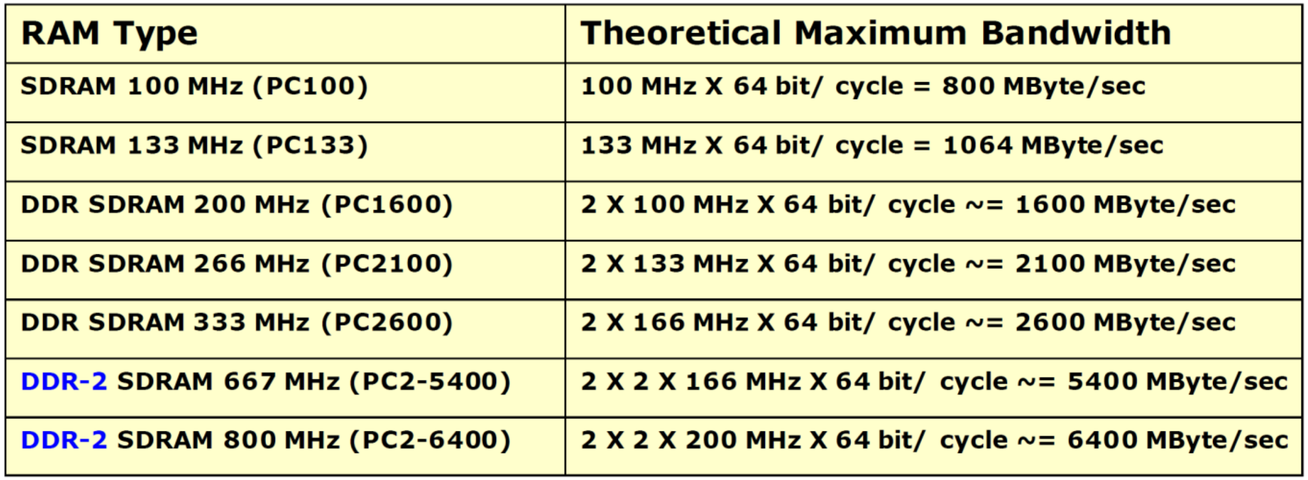
- normal SDRAMs only operate once per clock cycle
- Double Data Rate (DDR) SDRAM transfers data on both clock edges
- DDR-2 (4x basic memory clock) and DDR-3 (8x basic memory clock) are in the market.
- They offer increased storage capacity, lower power and faster clock speeds.
- For example, DDR2 can operate at clock frequencies of 400 and 800 MHz.
- The latest version is DDR-4.
- A DDR4-3200 can do 3200 million transfers per second/ 1600 MHz clock

Performance of SDRAM





1 Hertz
1 Cycle per second



RAM Type	Theoretical Maximum Bandwidth
SDRAM 100 MHz (PC100)	100 MHz X 64 bit/ cycle = 800 MByte/sec
SDRAM 133 MHz (PC133)	133 MHz X 64 bit/ cycle = 1064 MByte/sec
DDR SDRAM 200 MHz (PC1600)	2 X 100 MHz X 64 bit/ cycle ~= 1600 MByte/sec
DDR SDRAM 266 MHz (PC2100)	2 X 133 MHz X 64 bit/ cycle ~= 2100 MByte/sec
DDR SDRAM 333 MHz (PC2600)	2 X 166 MHz X 64 bit/ cycle ~= 2600 MByte/sec
DDR-2 SDRAM 667 MHz (PC2-5400)	2 X 2 X 166 MHz X 64 bit/ cycle ~= 5400 MByte/sec
DDR-2 SDRAM 800 MHz (PC2-6400)	2 X 2 X 200 MHz X 64 bit/ cycle ~= 6400 MByte/sec

- Bandwidth comparison. However, due to latencies, SDRAM does not perform as good as the figures shown.

DRAM Performance Factors

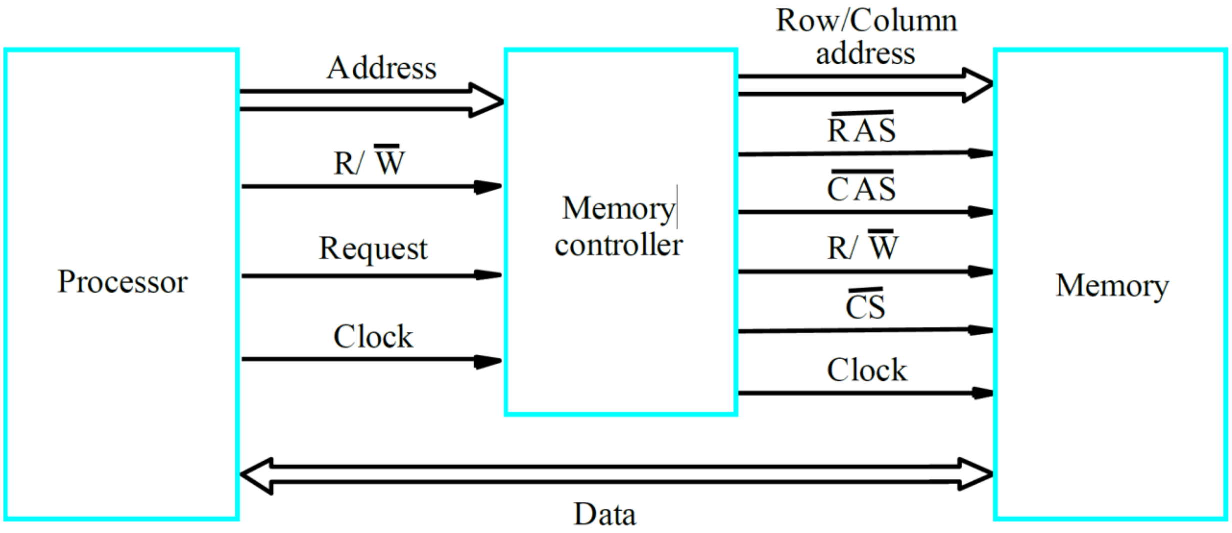
- Row buffer
  - Allows several words to be read and refreshed in parallel
- Synchronous DRAM
  - Allows for consecutive accesses in bursts without needing to send each address
  - Improves bandwidth
- DRAM banking
  - Allows simultaneous access to multiple DRAMs
  - Improves bandwidth

SRAM v.s. DRAM

- Static RAM (SRAM)
  - Capable of retaining the state as long as power is applied.
  - They are fast, low power (current flows only when accessing the cells) but costly (require several transistors), so the capacity is small.
  - Implement Level 1 cache and Level 2 cache inside a processor, of size 3 MB or more.
- Dynamic RAM (DRAM)
  - store data as electric charge on a capacitor.
  - Charge leaks away with time, so DRAMs must be refreshed.
  - In return for this trouble, much higher density (simpler cells).

Memory Controller

- A memory controller is normally used to interface between the memory and the processor.
- DRAMs have a slightly more complex interface as they need refreshing and they usually have time-multiplex signals to reduce pin number.
- SRAM interfaces are simpler and may not need a memory controller.



RAS (CAS) = Row (Column) Address Strobe; CS = Chip Select

Memory Controller

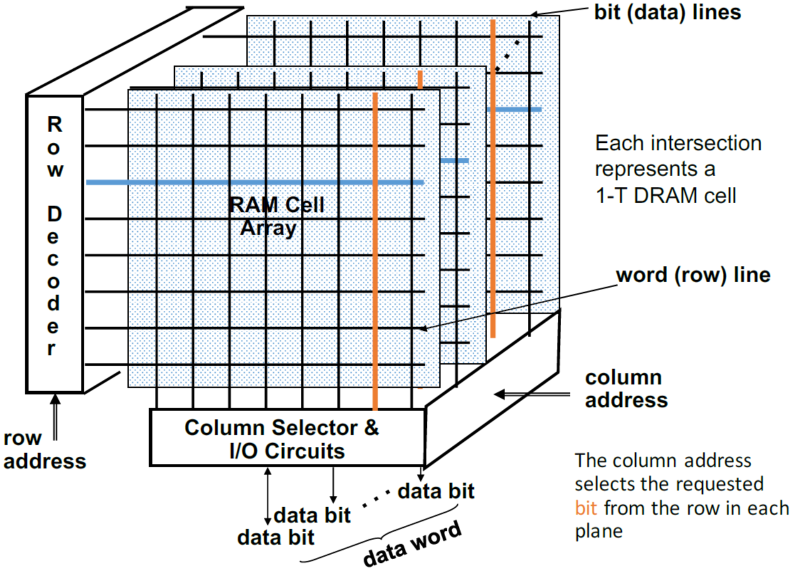
- The memory controller accepts a complete address and the R/W signal from the processor.
- The controller generates the RAS (Row Access Strobe) and CAS (Column Access Strobe) signals.
- The high-order address bits, which select a row in the cell array, are provided first under the control of the RAS (Row Access Strobe) signal.
- Then the low-order address bits, which select a column, are provided on the same address pins under the control of the CAS (Column Access Strobe) signal.
- The right memory module will be selected based on the address. Data lines are connected directly between the processor and the memory.
- SDRAM needs refresh, but the refresh overhead is only less than 1 percent of the total time available to access the memory.

Memory Module Interleaving

- Processor and cache are fast, main memory is slow.
- Try to hide access latency by interleaving memory accesses across several memory modules.
- Each memory module has own Address Buffer Register (ABR) and Data Buffer Register (DBR)

Memory Module Interleaving

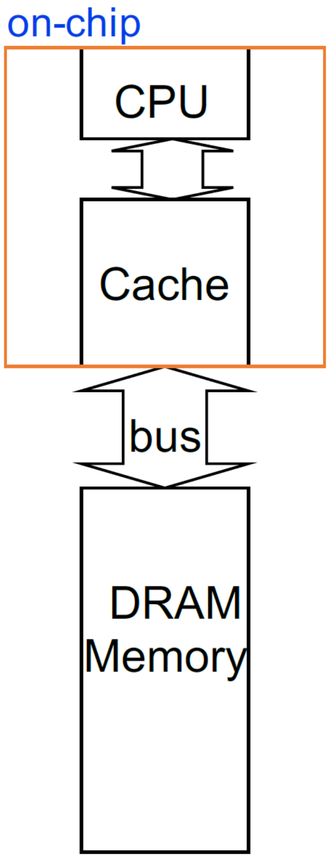
- Two or more compatible (identical the best) memory modules are used.
- Within a memory module, several chips are used in “parallel”.
- E.g. 8 modules, and within each module 8 chips are used in “parallel’. Achieve 8 *8 = 64-bit memory bus.
- Memory interleaving can be realized in technology such as “Dual Channel Memory Architecture”.



Each intersection represents a 1-T DRAM cell

The column address selects the requested bit from the row in each plane

Non-Interleaving v.s. Interleaving



on-chip

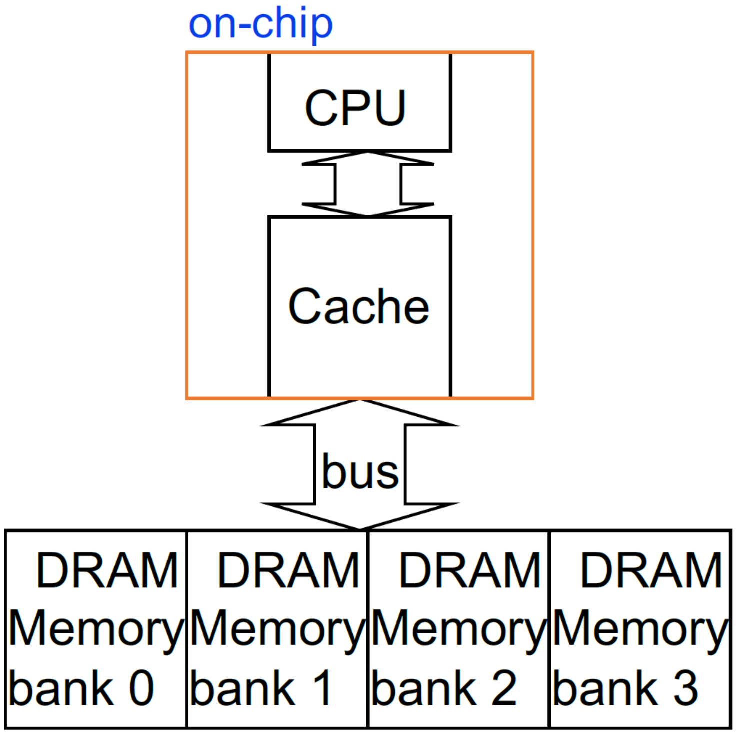
CPU

Cache

bus

DRAM Memory

(a) Non-Interleaving



on-chip

CPU

Cache

bus

DRAM Memory bank 0

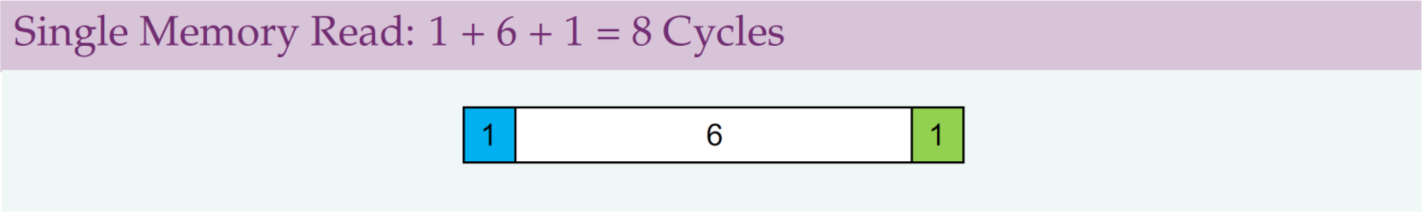
DRAM Memory bank 1

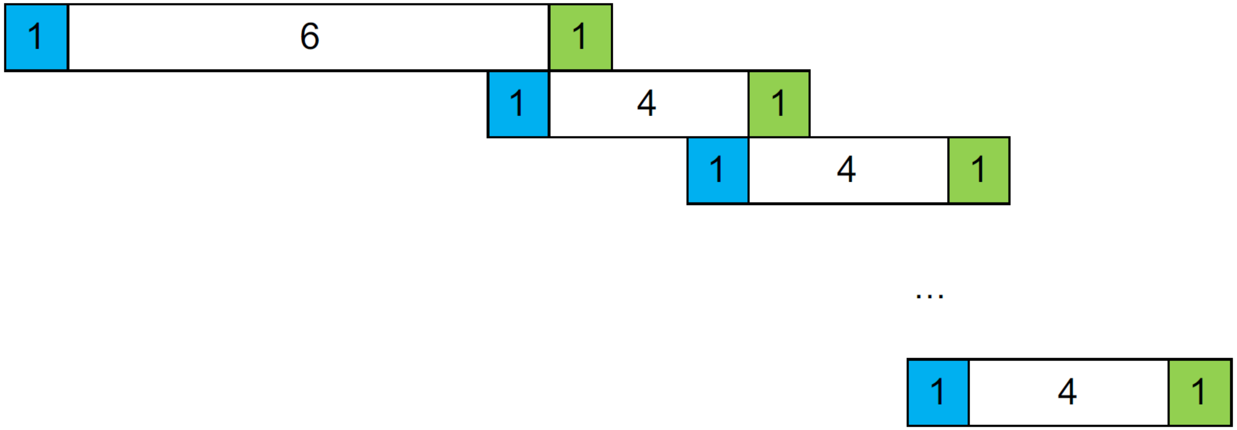
DRAM Memory bank 2

DRAM Memory bank 3

(b) Interleaving

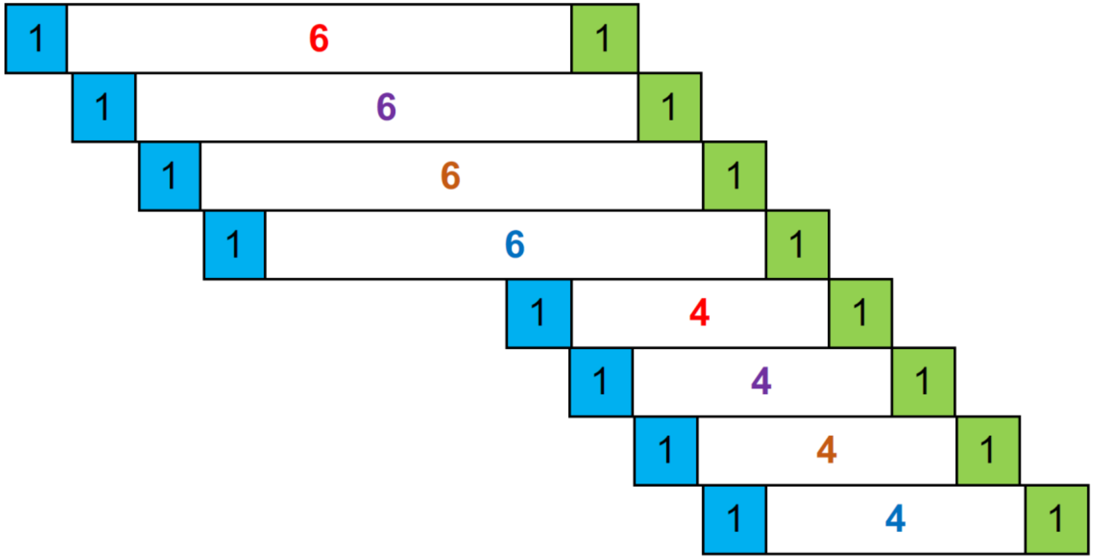
- Suppose we have a cache read miss and need to load from main memory
- Assume cache with 8-word block, i.e., cache line size = 8 words (bytes)
- Assume it takes one clock to send address to DRAM memory and one clock to send data back.
- In addition, DRAM has 6 cycle latency for first word
- Good that each of subsequent words in same row takes only 4 cycles


Single Memory Read: 1 + 6 + 1 = 8 Cycles



- First word DRAM needs 6 cycle (same as single memory read)
- All subsequent words DRAM needs 4 cycle
- Non-overlappings in cache access
- Assumption: all words are in the same row
- Non-Interleaving Cycle# : 1 + 1 * 6 + 7 * 4 + 1 = 36

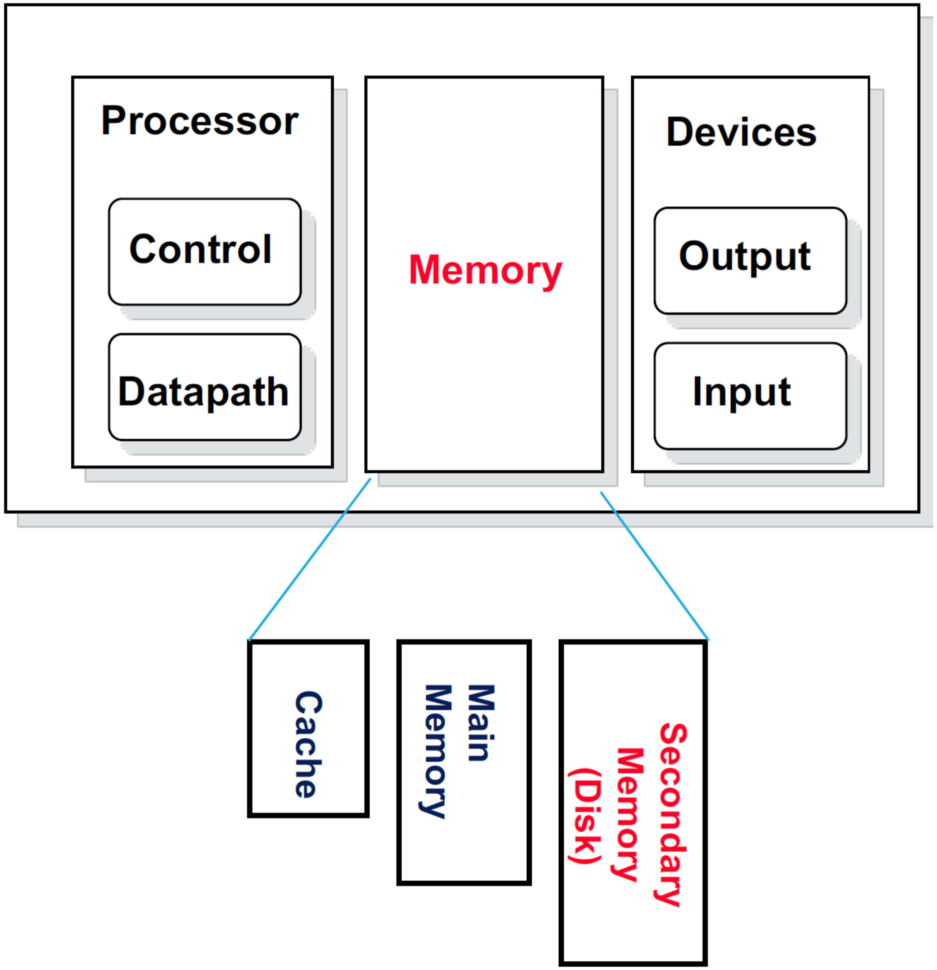
Example: Four Module Interleaving



Interleaving Cycle#

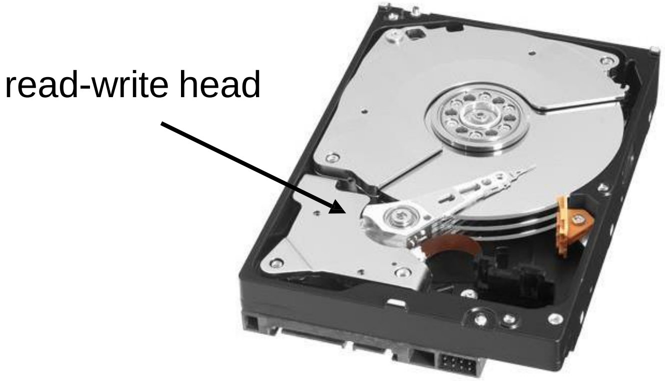
$$1 + 6 + 1 \times 8 = 15$$

Components of A Computer



Secondary Memory – Magnetic Hard Disk

  - Nonvolatile, rotating magnetic storage
  - Lowest level memory: slow; large; inexpensive

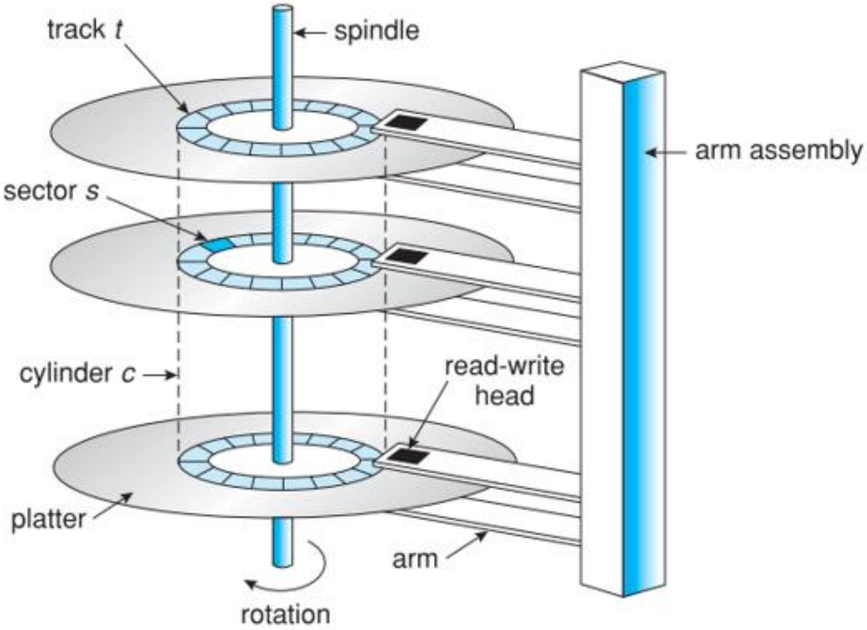


- Data are stored in sectors

- Access to a sector involves
  - Queuing delay
  - Seek time

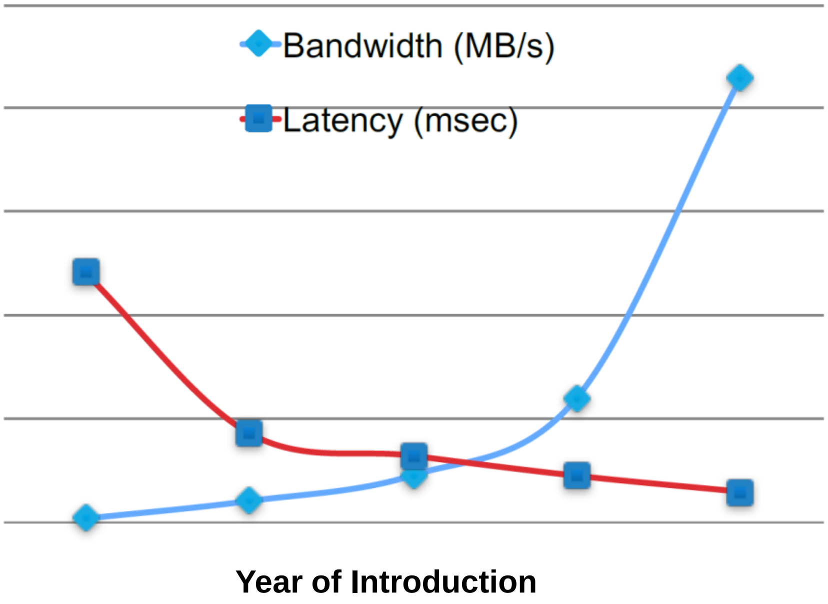
    the time for the disk to move the heads
    to the cylinder containing the desired sector
  - Rotational latency

    The time waiting for the disk to rotate the
    desired sector to the disk head.
  - Data transfer
  - Controller overhead



Magnetic Disk (Cont.)

- Latency: average seek time plus the rotational latency
- Bandwidth: peak transfer time of formatted data from the media (not from the cache)



- In the time the bandwidth doubles, latency improves by a factor of only around 1.2

Read-Only Memory (ROM)

- Memory content fixed and cannot be changed easily.
- Useful to bootstrap a computer since RAM is volatile (i.e. lost memory) when power
- removed.
- We need to store a small program in such a memory, to be used to start the process of loading the OS from a hard disk into the main memory.
- **PROM/EPROM/EEPROM**
  - Programmable ROM (PROM): A type of ROM that is programmed using high voltages.
  - Erasable programmable ROM (EPROM): A type of ROM that is programmed using high voltages and content can be erased by exposure to ultraviolet light for about 20 minutes.
  - Electrically erasable programmable ROM (EEPROM): It can be erased and reprogrammed several times while enabling the erase and writing of only one location at a time
- Flash memory is an updated version of EEPROM that allows numerous memory locations to be changed at the same time.

- *A bootstrap program is* ***the first code that is executed when the computer system is started****.*

Secondary Memory - Flash

- Flash Storage
  - Nonvolatile semiconductor storage
  - 100x ~ 1000x faster than Hard Disk
    - No rotation/seek time required
  - Wear leveling to overcome wear out problem
- FLASH Memory
  - Flash devices have greater density, higher capacity and lower cost per bit.
  - Can be read and written
  - This is normally used for non-volatile storage
  - Typical applications include cell phones, digital cameras, etc.

FLASH Cards

- Flash cards are made from FLASH chips
- Flash cards with standard interface are usable in a variety of products.
- Flash cards with USB interface are widely used – memory keys.
- Larger cards may hold 32GB. A minute of music can be stored in about 1MB of memory, hence 32GB can hold 500 hours of music.

Memory Hierarchy Design

Four Questions for Memory Hierarchy

Q1: Where to place a block in upper level?
(*Block placement*)
Anywhere, in a single specific place, in one out of several specific places

Q2: How to find a block in a the upper level?
(***Block identification***)

Q3: Which block should be replaced on a miss in upper level?
(***Block replacement***)
Replacement policy

Q4: What happens on a write in upper level?
(***Write strategy***)

Q4: What happens on a write?

- Write policies
  - Write–through—The information is written to both upper and lower level
    - easier to implement
    - the lower level has the most current copy of the data
    - Data consistency, coherence
  - Write–back—The information is only written to the upper level; the modified block is written to the lower level only when it is replaced
    - uses less bandwidth, since multiple writes within a block only requires one write to lower level
    - a read miss (which causes a block to be replaced and therefore) may result in writes to lower level
- A block in a write–back upper level can be either clean or dirty, depending on whether the block content is the same as that in lower level

Summary of Different Memories

Types	Access & Control	Volatility	Technology	Main purpose
Register	Controlled by CPU via instructions	Volatile	Gated D latch	Support ISA for fast execution
Cache	Controlled by hardware; accessed transparently	Volatile	SRAM	Hide latency of main memory
Main memory	Controlled by CPU via instruction	Volatile	DRAM	Store working code and data
Secondary storage	Controlled by OS via system calls	Non-volatile	Disk, flash	Store persistent program files and data files

Conclusions

- Processor usually runs much faster than main memory
- Common RAM types:
  - SRAM, DRAM, SDRAM, DDR SDRAM
- Principle of locality: Temporal and Spatial
  - Present the user with as much memory as is available in the cheapest technology.
  - Provide access at the speed offered by the fastest technology.
- Memory hierarchy:
  - Register->Cache-> Main Memory-> Disk->Tape