Fundamentals of System Hardware

Overview

- Definition of a computer
- Types of computers
- What’s inside a computer (macro view)
- What each component does
- Commonalities between components
- How components communicate
- How the CPU works
- The memory hierarchy
- Hard disks
- Networking

Definition

- The computer is an electromechanical device which takes input, does processing and produces output.

Types of computers

- Mainframe
- Server
- Desktop
- Laptop
- Tablet
- Portable phone

Inside a computer

What’s common between them?

- All computers have:
  - at least one Central Processing Unit (CPU) which is the “brain” of the computer.
  - Main memory where code and data is stored temporarily
  - Secondary storage where information is stored permanently
- Most computers will have:
  - A video graphic controller where images can be rendered for display on a screen
  - A network interface for communications
  - Peripheral interfaces (USB, Thunderbolt, Firewire, SCSI, etc)

Communications between the devices

- Internal communications in a machine is done via a “bus.”
- A bus is a physical pathway for communication between two or more devices
- The system bus is the main pathway between the CPU and main memory, but also carries data to and from Input and Output (IO) devices.

The CPU

- The CPU is the “brain” of the computer
- It is a single piece of silicon in the form of a chip.
- This is the only location where code is actually executed in the system.
- The CPU only runs “machine language” code.
- The CPU operates on a “fetch-decode-execute” cycle.
- Each type of CPU has its own set of “instructions” which is understands.
- Each CPU has a small amount of memory, call “registers” which it uses to perform operations and store results.
- A CPU ay have a “cache” memory to perform more quickly.

Machine-Language

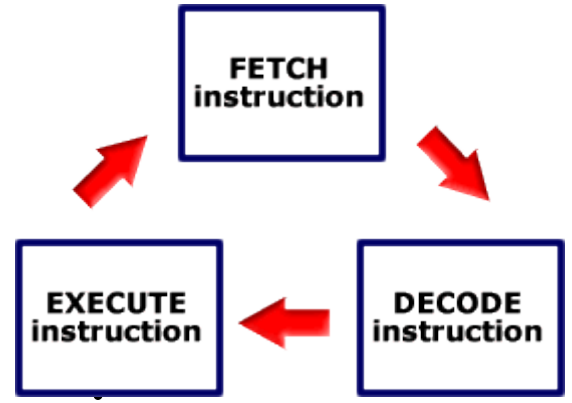
- Computers can only understand very basic commands like
  - Move
  - Add
  - Subtract
  - Multiply
  - Compare
  - Jump
  - etc
- The designer of the CPU puts the capability to perform these operations in the physical chip.

Instruction set

- The designers of the CPU create a set of instructions that the CPU can perform.
- This set of instructions, usually as small as 100, can each be represented by a numeric value.
- When the CPU receives a particular instruction, it performs that task.

Instruction	Opcode
ADD	0x00
AND	0x20
CMP	0x38
INT	0xCD
JMP	0xE9

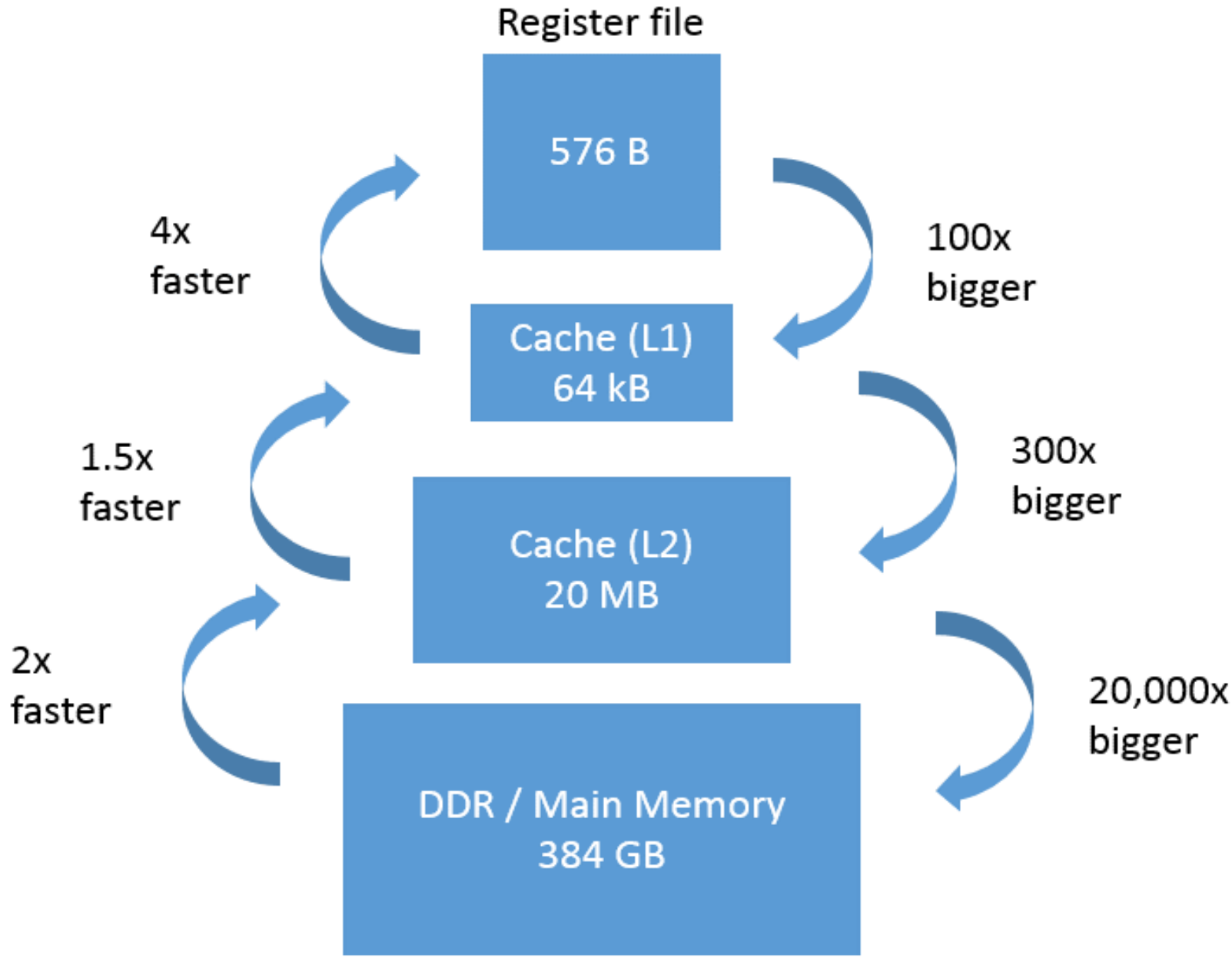
Fetch-Execute Cycle



- The CPU performs a fetch to move the instruction from main memory into the CPU (specifically into an instruction register).
- It then decodes the instruction, also moving in any additional data that might be necessary with that instruction
- It then executes that instruction
- This process repeats with the next instruction in the sequence
- This whole process can take as little as about 0.00000001 seconds or about 10 nanoseconds. Meaning the CPU can process millions of instructions per second.

Memory

- The Instruction, and all the data, has to come from somewhere.
- In order for code to be executed, it has to be in a register built into the CPU
- Why not just store everything in registers? They are EXPENSIVE!
- As memory gets faster, it tends to get more expensive. So we have a hierarchy

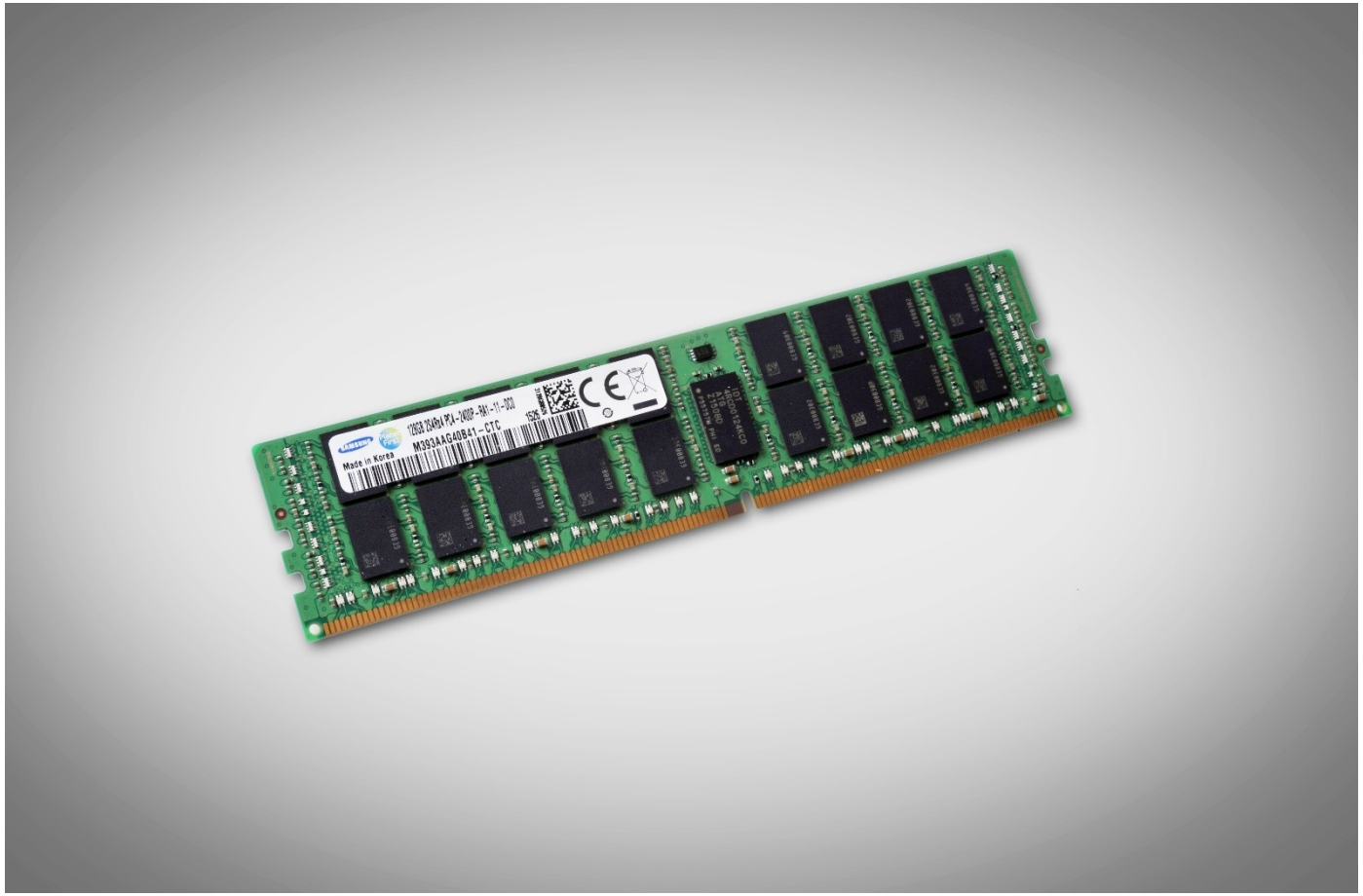

Register file
576 B
4x
faster
100x
bigger
Cache (L1)
64 kB
1.5x
faster
300x
bigger
Cache (L2)
20 MB
2x
faster
20,000x
bigger
DDR / Main Memory
384 GB

The hierarchy

- Registers (Nanosecond access time, but size measured in bytes)
  - Instructions can be executed only here
- Cache (Nanosecond access time, size measured in megabytes)
  - Useful for the processor designers, not terribly useful for programmers
- RAM (10+ nanosecond access time, size measured in gigabytes)
  - This is where all of your code and data will be stored temporarily
  - This is known as volatile memory because it is all lost when the computer is off
- Secondary Storage(10 millisecond access time, size measured in terabytes)
  - Permanent storage, your hard drive.
- Tierterary storage (offline, size measured only by your imagination)
  - Used for backup or other information that doesn’t need to be immediately accessible

RAM

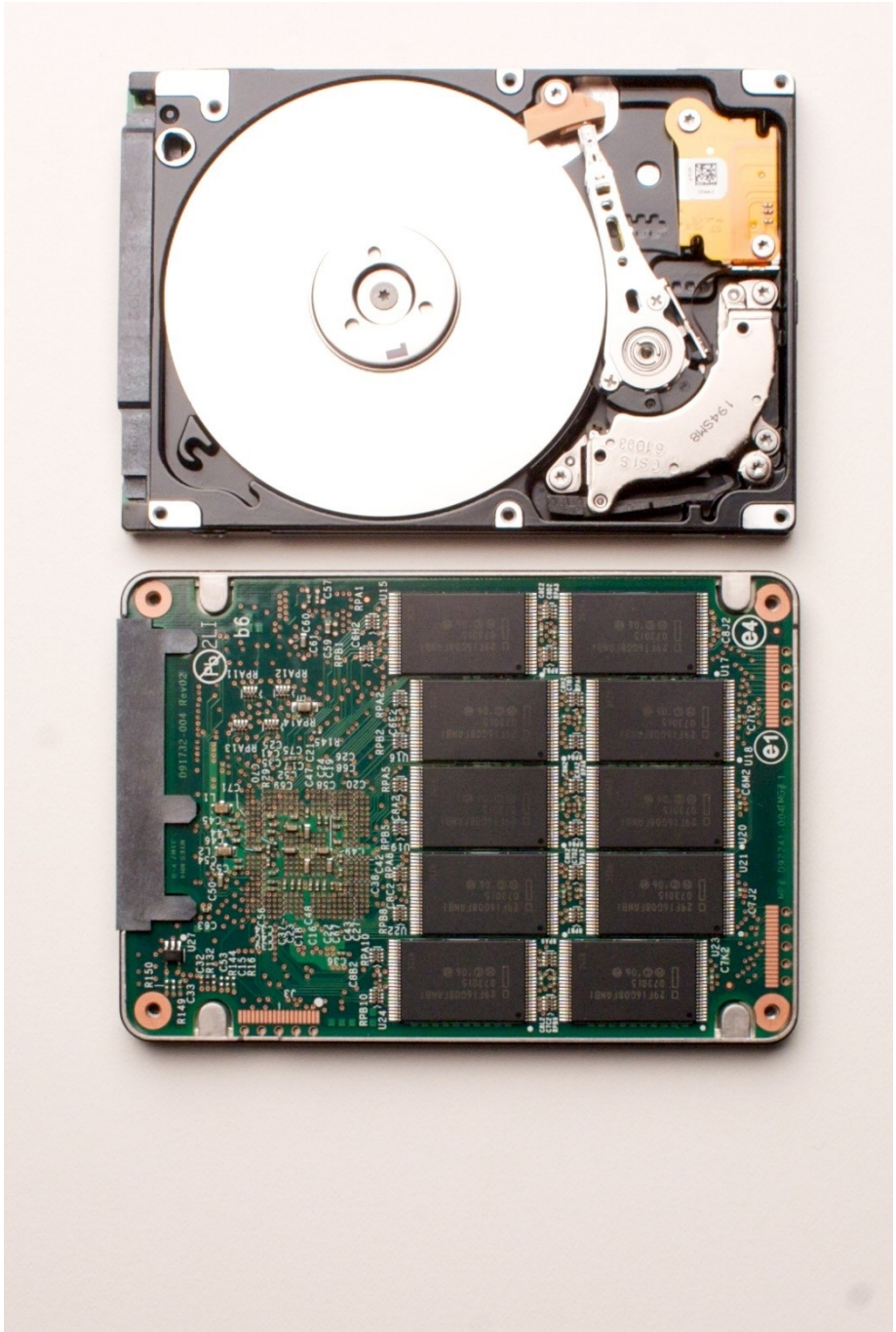
- Known as RANDOM Access Memory because any place in it can be accessed in the same amount of time (which is not necessarily true for older secondary storage).
- This areas of memory is broken down into bytes, with each byte being able to be accessed independently of the others
- When the computer is turned off, everything in RAM is lost.
- When running a program, all the machine language instructions are brought into RAM and, one-by-one, pulled into the CPU by the fetch and execute cycle.


Made in Korea
M393A4G40B41-CTC

Secondary Storage

- Secondary storage can usually be broken down into two types
  - Hard disk drive (HDD) also known as “spinning” drives.
    - Contain multiple magnetic material discs which rotate together at a constant velocity
    - Contain read heads which move to different radii on the disk
    - Allow the system to access any position via it’s three dimensional polar coordinates
    - Accessing first the innermost radius then the outermost radius takes significantly longer than two adjacent radii.
    - Size is usually measured in terabytes
  - Solid State Disks
    - Contain a number of chips like USB flash drives.
    - Data is stored, electrically, in these chips
    - All data can be access in the same amount of time
    - Due to cost, these drives a smaller than HDDs but perform much faster.

HDD vs. SSD



Networking

- The world is now globally connected
- Data can come from anywhere as long as the data and we are connected to the same network.
- Networks are interconnected via the Internet

Physical connections

- Types of connections
  - Copper
  - Fiber
  - Wireless
- Protocols
  - How do we know when to start sending data?
  - How do we know who its from?
  - How do we know who its going to?
- Types of physical connections
  - Ethernet
  - Wifi (802.11)
  - ATM

Packets

- Unlike a telephone, true connections on a network are rare.
- Data is sent from a program to another program indirectly by using multiple protocols.
- Generally there will be protocols for:
  - Each application
  - Each logical network
  - Each physical network
- In order to send data it is encapsulated into a packet and given to the application protocol, which then adds a header for information for the receiver to understand what's being sent.
- When the packet moves to the logical protocol, it too adds a header on top of the application layer's header.

Data-link Network Application DATA

Layers we commonly use

- Application
  - HTTP – Hyper Text Transport Protocol
  - SMTP – Simple Mail Transport Protocol
  - IMAP – Internet Mail Access Protocol
- Logical (more formally called Network)
  - Usually broken down into two layers
    - Connection oriented vs. Connection less deals with the ordering and guarantee of delivery of packets
      - UDP
      - TCP
    - Global delivery of packets
      - Internet Protocol – IP
- Physical
  - Often adds a header and footer (to indicate the end)
  - Mostly concerned with local addressing and how to deal with eccentricities of the physical medium
  - Ethernet
  - 802.11