Abstraction and Computing Systems

Textbook Chapter 1
Computing machines are everywhere

• General purpose
  – Servers, desktops, laptops, tablets, smart phones, etc.

• Special purpose
  – Cash registers, ATMs, games, telephone switches, etc.

• Embedded
  – Cars, hotel doors, printers, VCRs, industrial machinery, medical equipment, etc.
Computing machines: distinguishing features

• Speed
• Cost
• Price/performance
• Ease of use, software support & interface
• Scalability
• Power
• Size
Computing System

Samsung Galaxy S5
1st Very Important Idea

• Universal Computational Devices
  – Given enough time and memory, all computers are capable of computing exactly the same things
  – Irrespective of speed, size, or cost

• Turing’s Thesis
  – Every computation can be performed by some Turing Machine – a theoretical universal computational device
Alan Turing’s original model

(1912-1954)
A Turing Machine

Also known as a *Universal Computational Device*: a theoretical device that accepts both input data and instructions on how to operate on the data.
2nd Very Important Idea

• Problem Transformation
  – The ultimate objective is to transform a problem expressed in natural language into electrons running around a circuit

• This is computer science and computer engineering
  – A continuum that embraces software and hardware
Computer Architecture

- Problems
- Algorithms
- Language
  - Instruction Set Architecture
- Microarchitecture
- Circuits
- Devices
Computer Science

**Definition:** The study of algorithms and data structures to solve problems.

**Abstraction:** Use of level of abstraction in software design allows the programmer to focus on a critical set of problems without having to deal with irrelevant details.
Procedure or Function

```plaintext
int average (a, b)
begin
    int avg;
    avg = (a+b)/2;
    return (avg);
end

main ()
...
x = 4;
y = 2;
k = average (x,y);
print (%d, k);
...
```
**Compiler**: A computer program that translates code written in a high level language into an intermediate level abstract language.

**Assembler**: A computer program that translates code written in assembly language to the binary form that the CPU can execute.
Computer Engineering

**Definition:** The creative application of engineering principles and methods to the design and development of hardware and software systems.

**Abstraction:** Use of level of abstraction in hardware design allows the designer to focus on a critical set of problems without having to deal with irrelevant details.
Instruction Set Architecture (ISA)

**Definition:** Interface between a computer’s hardware and its software. Defines exactly what the computer’s instructions do, and how they are specified.
Central Processing Unit
The heart of computing systems

ca 1980
It took 10 of these boards to make a Central Processing Unit (CPU)

ca 2000
No wonder they called this CPU a microprocessor!
Motherboard: System
CPU: Package
SoC – System on a chip

800 PROCESSOR

Krait 400 CPU features 28HPm process technology superior XGHz+ performance

Adreno 330 for advanced graphics

Hexagon QDSP6 for ultra low power applications and custom programmability

Integrated LTE 802.11ac 3.0 and BT 4.0 offers broad array of high speed connectivity

MULTIメディア
Audio, Video and Gestures

KRAIT CPU
ADRENO GPU
HEXAGON DSP

CONNECTIVITY
4G LTE, WIFI, USB, BT and FM

CAMERA
DISPLAY/LCD
NAVIGATION

Support for up to 2560x2048 display
Mirocast 1080p HD support

Ultra HD Capture and Playback
DTS-HD and Dolby Digital Plus audio
Expanded Gestures

55MP with dual ISP

12at GPS with support for three GPS constellations
CPU: Microarchitecture

Intel Core i7
CPU: Die

Memory Controller

Core
Core
Queue
Core
Core

Shared L3 Cache

Intel Core i7
CPU: Die with graphics core

Intel® Core™ M Processor Die Map
14nm 2nd Generation Tri-Gate 3-D Transistors

Dual Core Die Shown Above

Transistor Count: 1.3 Billion
4th Gen Core Processor (Y series): 1.96B
** Cache is shared across both cores and processor graphics

Die Size: 82mm²
4th Gen Core Processor (Y series): 131mm²
Two recurring themes

1) Abstraction

– The notion that we can concentrate on one “level” of the big picture at a time, with confidence that we can then connect effectively with the levels above and below.

– Framing the levels of abstraction appropriately is one of the most important skills in any undertaking.
Two recurring themes

2) Hardware vs. Software

– On the other hand, abstraction does not mean being clueless about the neighboring levels.
– In particular, hardware and software are inseparably connected, especially at the level we will be studying.
What is Computer Organization?

There is a fundamentally wide gap between the intended behavior desired and the workings of the electronic devices that do the work.

Before the digital computers of today special purpose analog devices (mechanical, electrical, or electronic) were built for each desired behavior.
A general purpose computer is the bridge that links the desired behavior (application) and the basic building blocks (electronic devices).
Our computer model for now

CPU Interacts with the memory in 3 ways:
- fetches instructions
- loads the value of a variable
- stores the new value of a variable

Memory is capable of only 2 operations:
- reads – a load or a fetch
- writes – operation of storing the value of a variable