CS2100
COMPUTER ORGANISATION

Lecture #1

Introduction

From High-Level Languages to Computer Organisation
(AY2022/23 Semester 1)

Questions? Click here or scan QR.

https://app.sli.do/event/qVCWNryB45Bnh6p2HRfnFG

Question and Answer

- Q&A QR code appears at the bottom left corner of the screen.
- QR code **may be obscured in some slides**.
- Also accessible at: https://app.sli.do/event/qVCWNryB45Bnh6p2HRfnFG
- Q&A link closes after the recitation.
  - Questions will be answered during the recitation.
  - You may post additional questions during the recitation:
    - If we are unable an answer a question during the recitation, we will do so on the Canvas forum.

Blended Learning Format

- New format that we are piloting this semester.
  - Lecture slides and videos will be uploaded in advance.
  - We meet every Tuesday face-to-face and over Zoom (hybrid) to answer your questions and do additional exercises.
    - First meeting on **16 August 2022.**
    - 4 pm to 6 pm, I3-Auditorium. Pre-registration required due to limited space.
    - Alternatively, attend online at:

https://nus-sg.zoom.us/j/89293172156?pwd=unlNkJJJnF8VS6NfqaN6HUOD9Ihr5.1

Meeting ID: 892 9317 2156

Passcode: 949665

Blended Learning Format

- New format that we are piloting this semester.
  - Lecture slides and videos will be uploaded in advance.
  - We meet every Tuesday face-to-face and over Zoom (hybrid) to answer your questions and do additional exercises.
    - You MUST read the slides and watch the videos in advance.
    - There will be 5 to 8 questions during the lectures for you to answer. Submissions close on Tuesday morning.
    - Totals 5% of your course grade.
  - What will we do on Tuesdays?
    - Answer questions on the Q&A.
    - Extra exercises to help you understand the materials better.

Lecture #1: Introduction

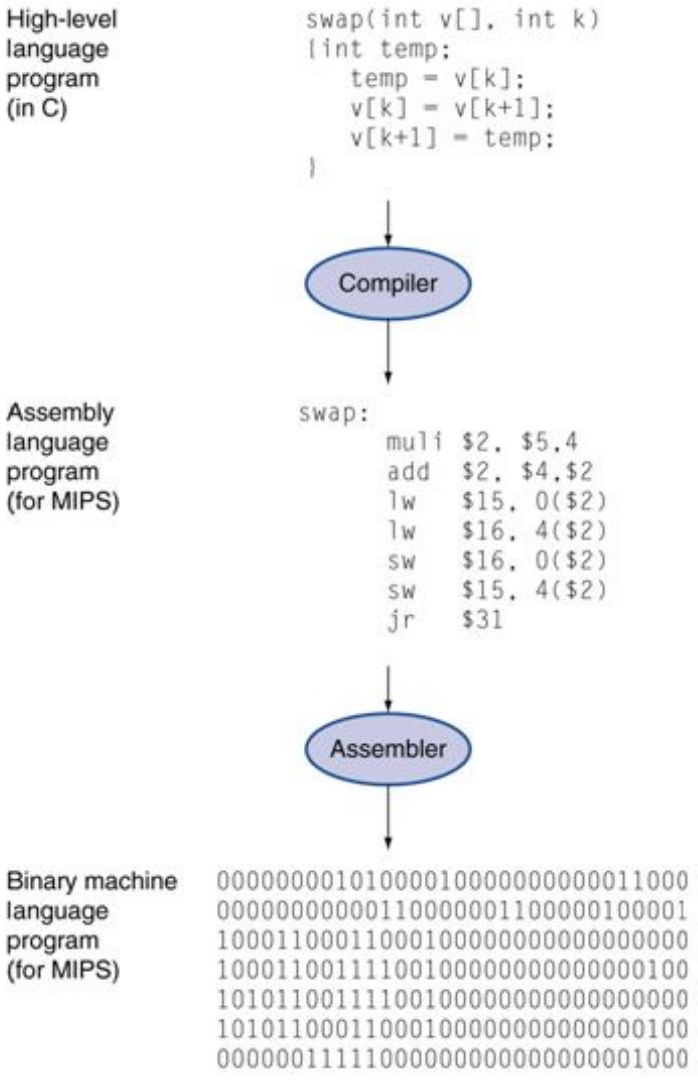
1. Programming Languages
2. C Programming Language
3. Abstraction
4. So, What is a Computer?
5. Why Study Computer Organisation?

1. Programming Languages

Programming language: a formal language that specifies a set of instructions for a computer to implement specific algorithms to solve problems.

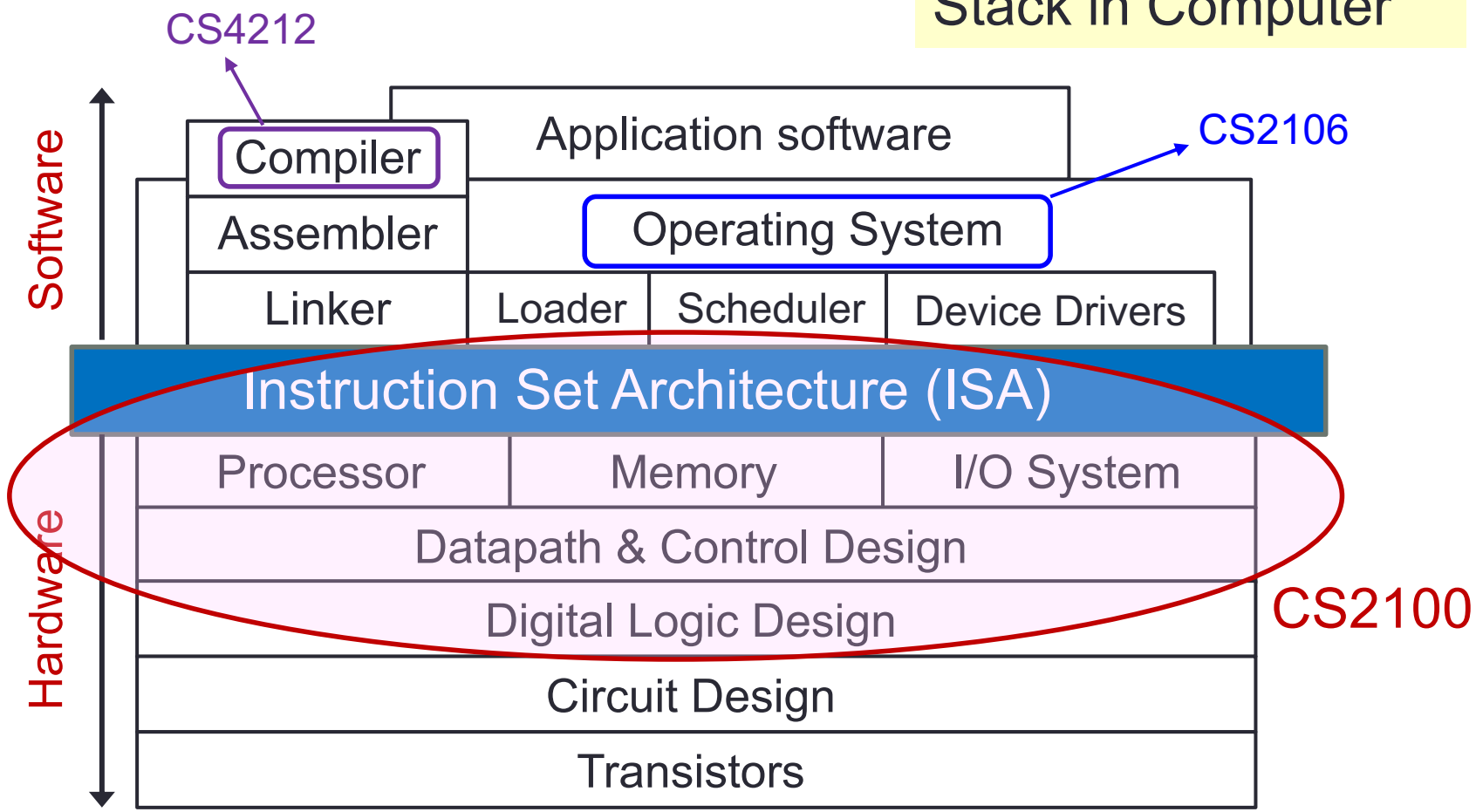
3. Abstraction (1/3)

- High-level language
  - Level of abstraction closer to problem domain
  - Provides productivity and portability
- Assembly language
  - Textual and symbolic representation of instructions
- Machine code (object code or binary)
  - Binary bits of instructions and data



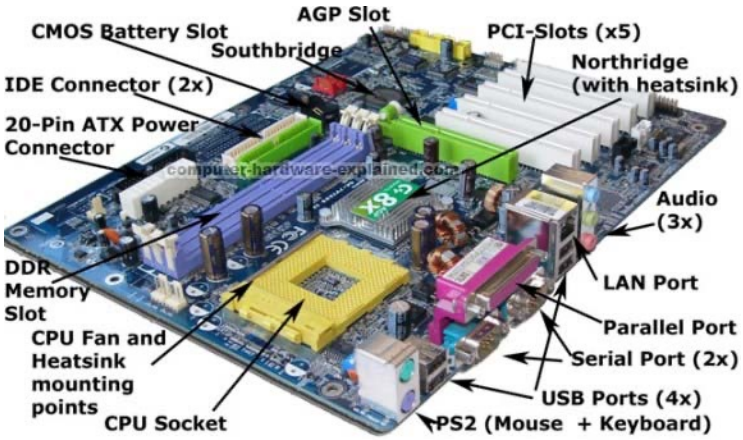
3. Abstraction Layers (2/3)

Hardware/Software Stack in Computer



4. So, What is a Computer? (4/6)

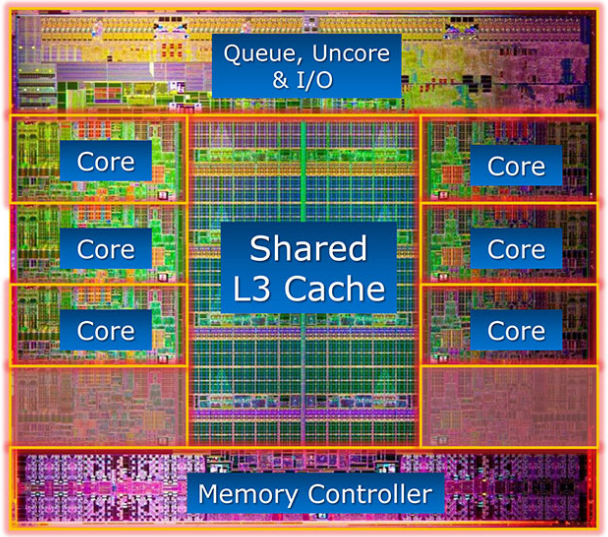
- PC motherboard



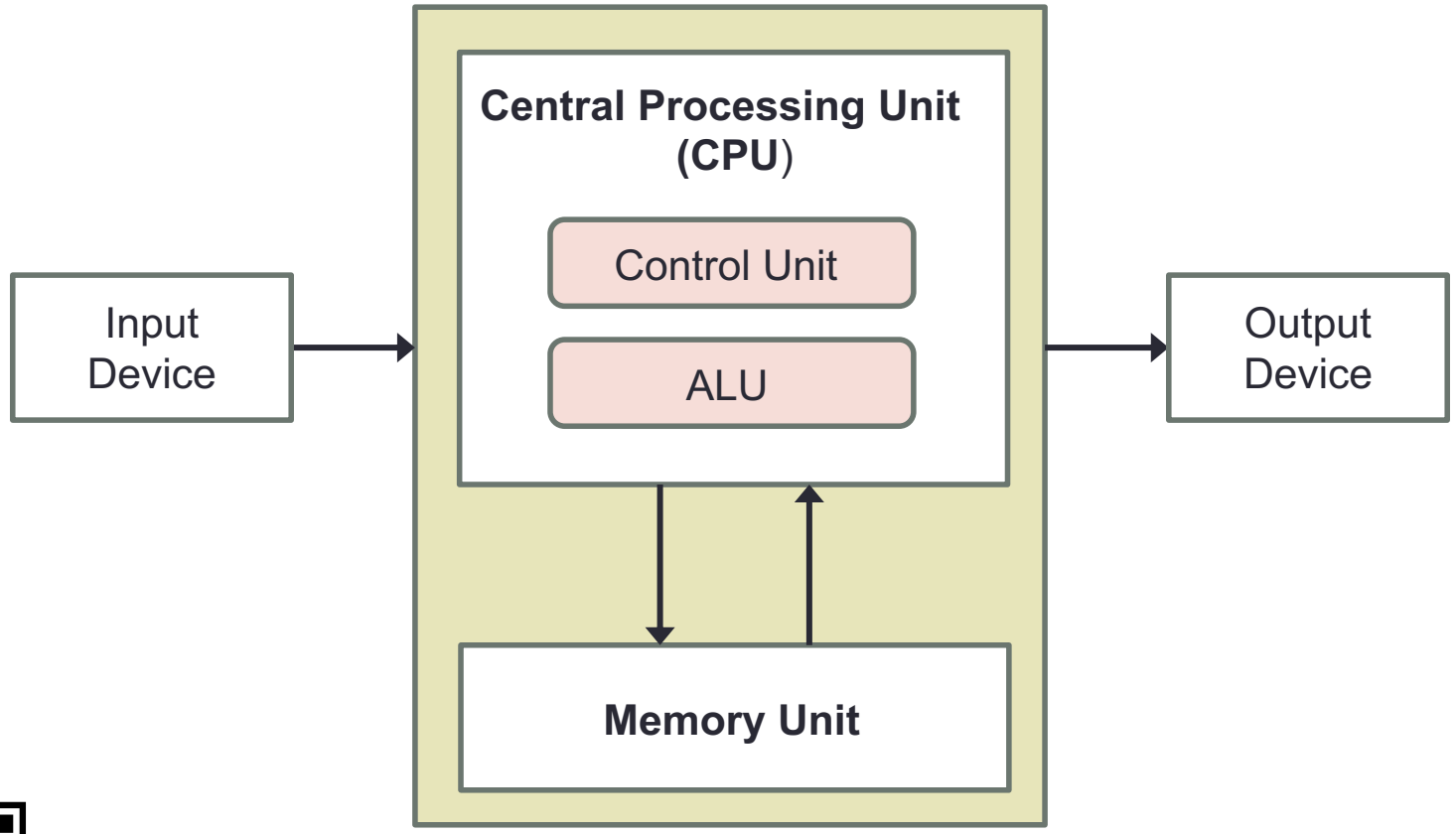
Credit: http://www.computer-hardware-explained.com/what-is-a-motherboard.html

- Intel i7 Processor

Intel® Core™ i7-3960X Processor Die Detail



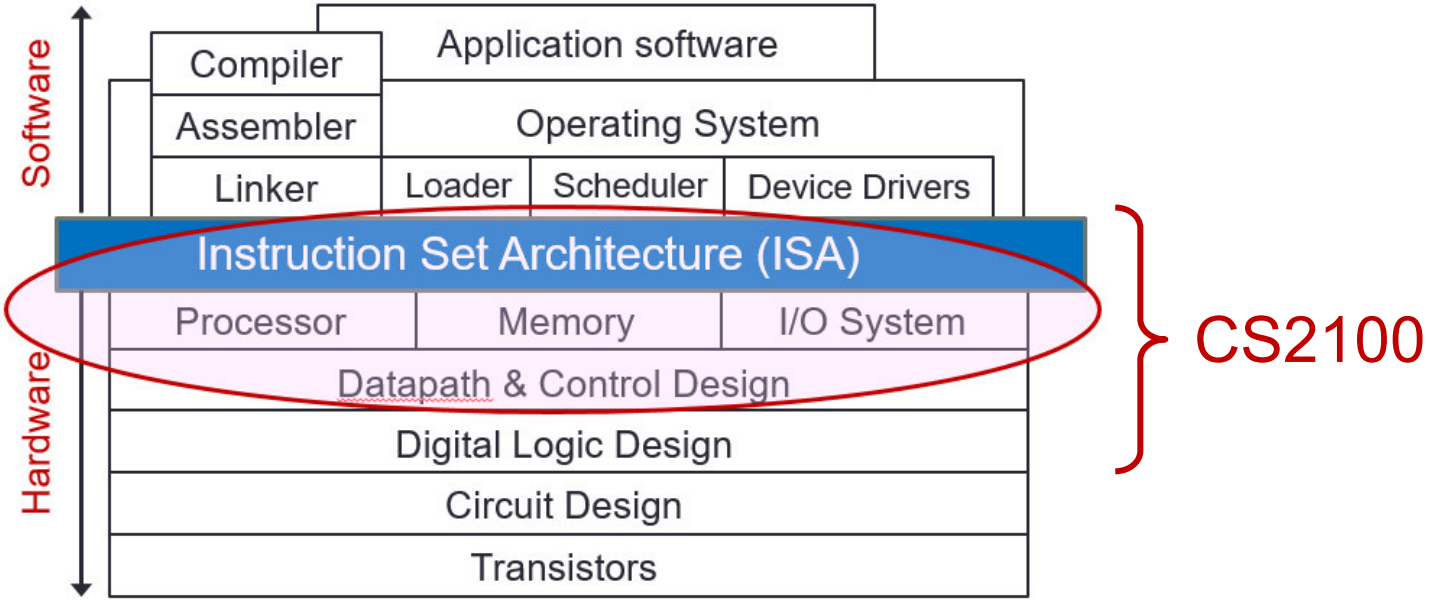
4. So, What is a Computer? (5/6)



ALU: Arithmetic/Logic Unit

5. Why Study Computer Organisation?

- Computer organisation is the study of internal working, structuring and implementation of a computer system.
- It refers to the level of abstraction above the digital logic level, but below the operating system level.



5. Why Study Computer Organisation?

(From user to builder)

- You want to call yourself a computer scientist/specialist.
- You want to build software people use.
- You need to make purchasing decisions.
- You need to offer "expert" advice.
- Hardware and software affect performance
  - Algorithm determines number of source-level statements (eg: CS1010, CS2030, CS2040, CS3230)
  - Language, compiler, and architecture determine machine instructions (COD chapters 2 and 3)
  - Processor and memory determine how fast instructions are executed (COD chapters 5, 6 and 7)
- Understanding performance (COD chapter 4)

End of File