CSE-312: Computer Architecture

Dr. Prasenjit Chanak

Assistant Professor

Department of Computer Science and Engineering
Indian Institute of Technology (BHU), Varanasi
UP, 221005

Topics for Today

- Course, Reference Book
- What do we study in this course?
- Why should this be studied?
- What is Computer Architecture ?
- How is the course structured?

Syllabus

- **Basic functional blocks of a computer**: CPU, memory, input-output subsystems, control unit. Instruction set architecture of a CPU - registers, instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set. Case study - instruction sets of some common CPUs, MIPS Instruction set (Introduction to 8085 Assembly Language Programming, Microprocessor Architecture and Microcomputer Systems, 8085/8086 Microprocessor Architecture and Memory Interfacing, Interfacing I/O Devices).
- **CPU control unit design**: hardwired and micro-programmed design approaches, Case study - design of a simple hypothetical CPU
- **Memory system design**: semiconductor memory technologies, memory organization
- Types and classification of architecture, Parallel computers, hypercube, systolic arrays models, Principles of scalable performance, Processor and memory hierarchy, Bus, Cache and shared memory, pipelining and super scalar techniques. Classification of architectures, Array processors, Vector processors, Vectorisation methods, supercomputers, Cray – cyber, etc.
- **Peripheral devices and their characteristics**
- **Pipelining**

Text Books

- Text Book
  - Computer Organization and Design: The Hardware/Software Interface, David A Patterson, John L. Hennessy, 4th Edition, Morgan Kaufmann, 2009
  - Computer Organization by V Carl Hamacher, Zvonks Vranesic, Safea Zaky, McGraw Hill, Vth Edition
  - Computer System Architecture by M Morris Mano, Prentice Hall of India, 2001
- Reference Book
  - Computer Architecture and Organization by William Stallings, PHI Pvt. Ltd., Eastern Economy Edition, Sixth Edition, 2003
  - Computer Architecture and Organization by John P Hayes, 3rd Ed. McGraw Hill, 2002

Evaluation Pattern

Evaluation Pattern	**Number of exam**	**Weightage (%)**
Quizzes	02 before mid semester and 02 after mid semester	20%
Mid semester	01	25%
Final Semester	01	40%
Final Semester Rea-time/real-time quiz		10%
Attendance, sincerity etc		5%

Course Objectives

- To learn-
  - How computers work, basic principles
  - How to analyze their performance (or how not to!)
  - How computers are designed and built
  - Issues affecting modern processors (caches, pipelines, etc.)

What is “Computer Architecture”

- Building architecture: Structural design
  (Civil Engg)

- Computer architecture :Circuit design
  (Electronics/CS Engg)

Abstraction

- Delving into the depths reveals more information
- An abstraction omits unneeded detail, helps us cope with complexity

Software Abstraction

C

```
int sum(int x, int y)
{
  int t = x+y;
  return t;
}
```

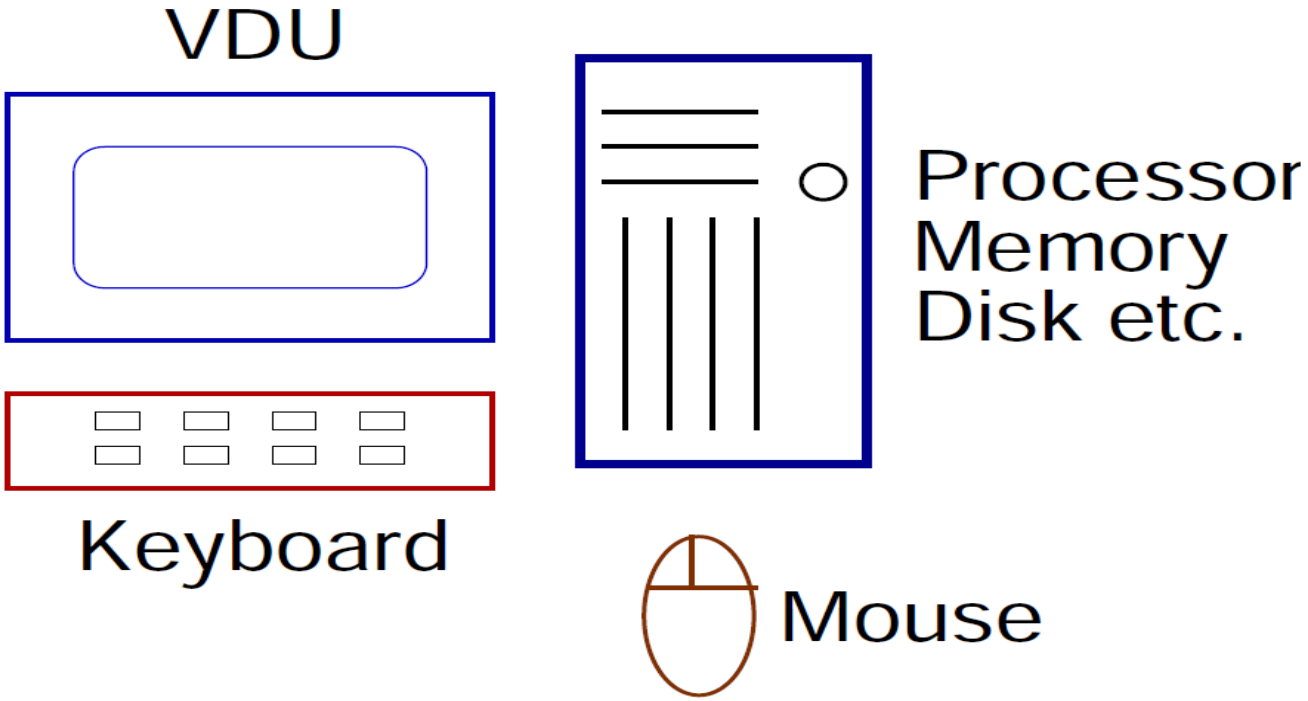
assembly

```
_sum:
      pushl %ebp
      movl %esp,%ebp
      movl 12(%ebp),%eax
      addl 8(%ebp),%eax
      movl %ebp,%esp
      popl %ebp
      ret
```

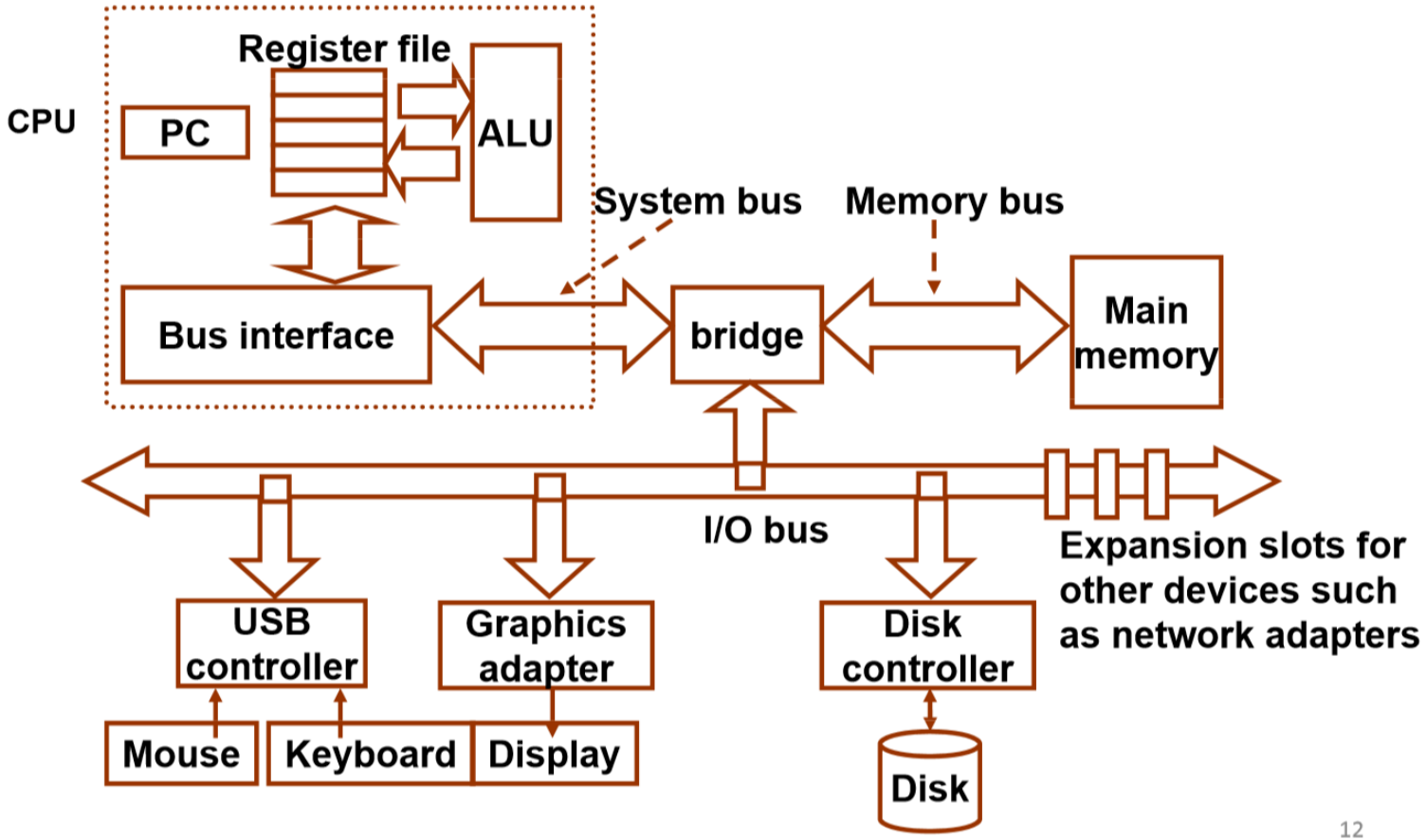
machine code

```
0x401040 <sum>:    0x55
                   0x89
                   0xe5
                   0x8b
                   0x45
                   0x0c
                   0x03
                   0x45
                   0x08
                   0x89
                   0xec
                   0x5d
                   0xc3
```

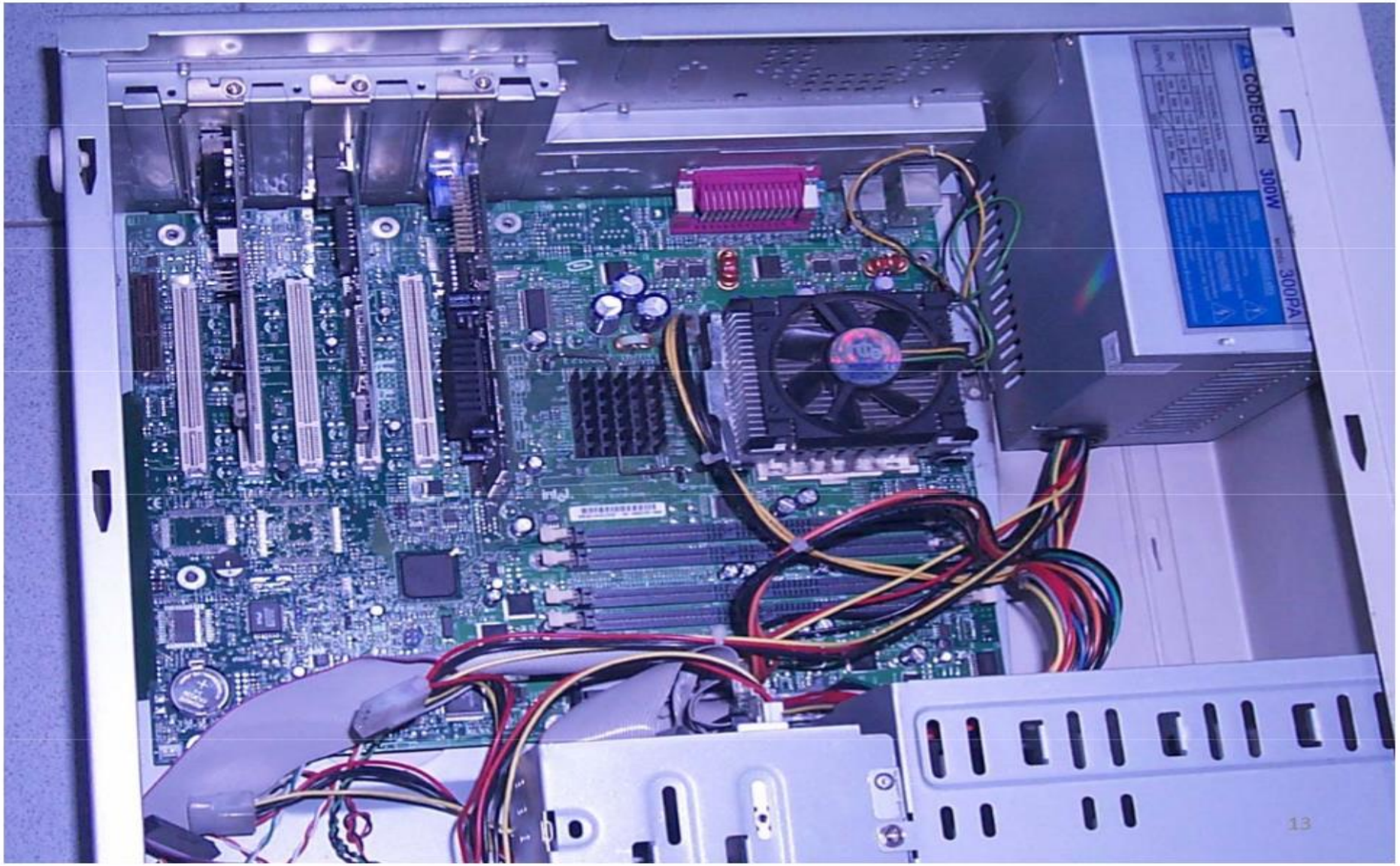
Computer System: Hardware Abstraction



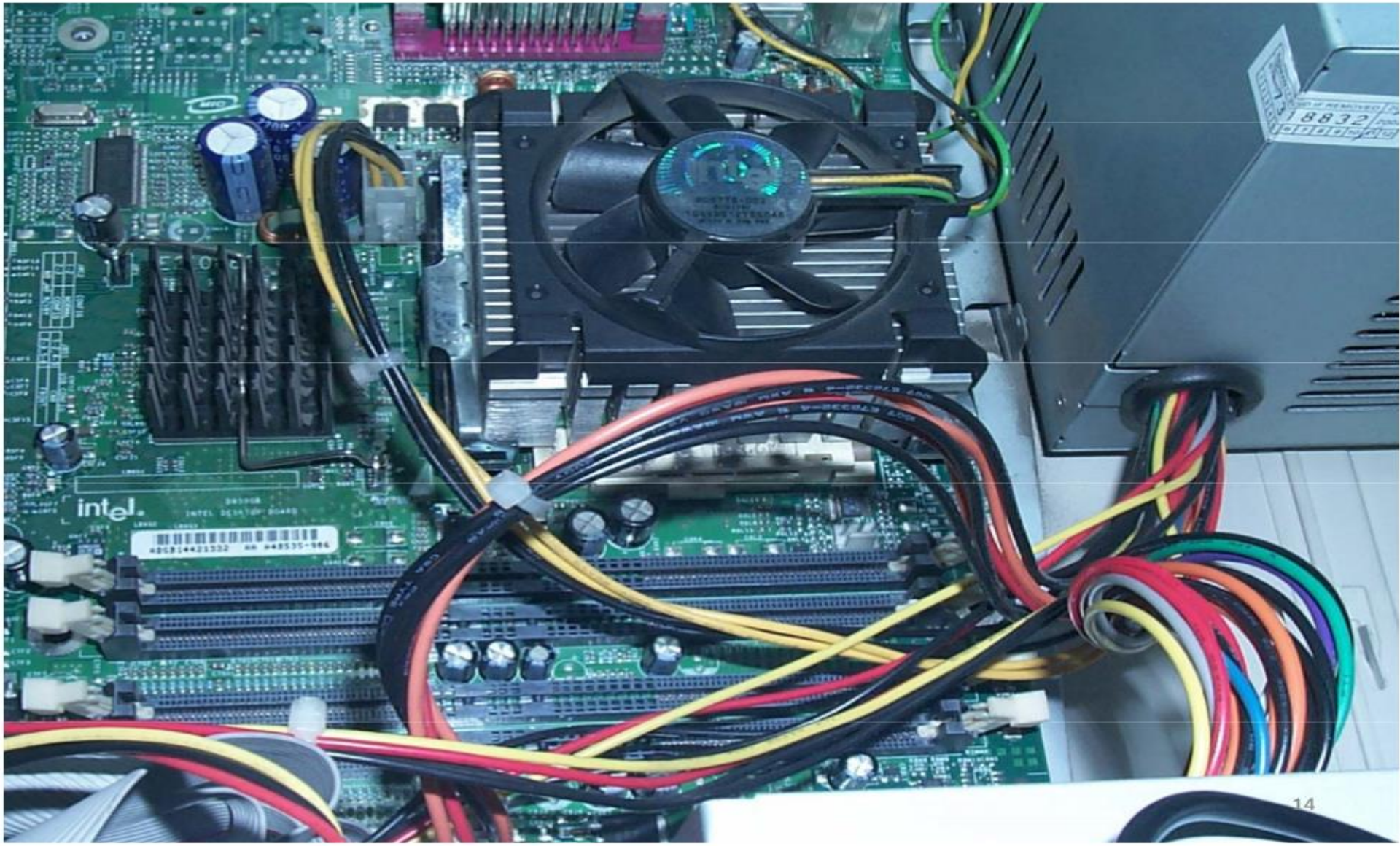
Hardware Abstraction



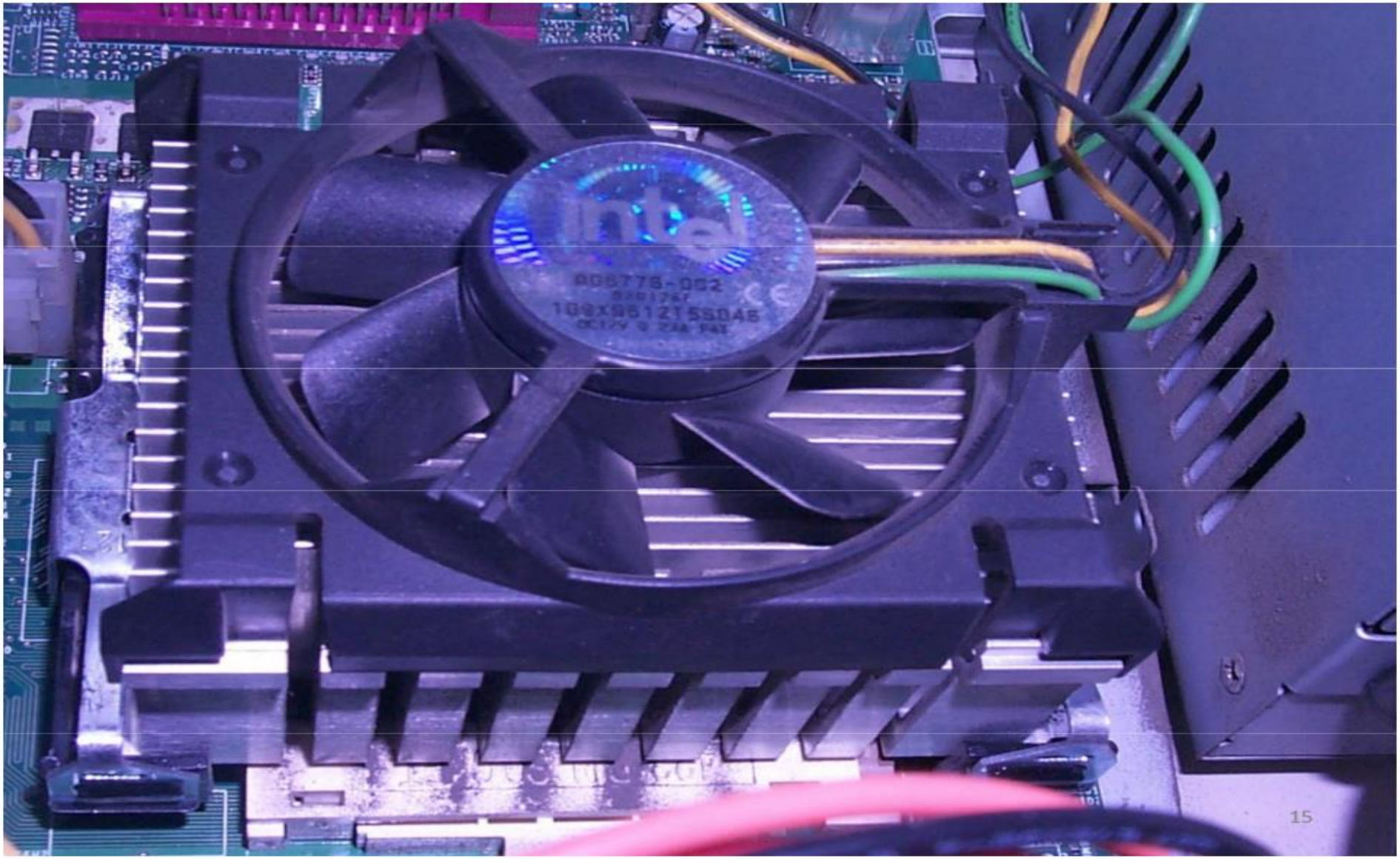
Hardware Abstraction



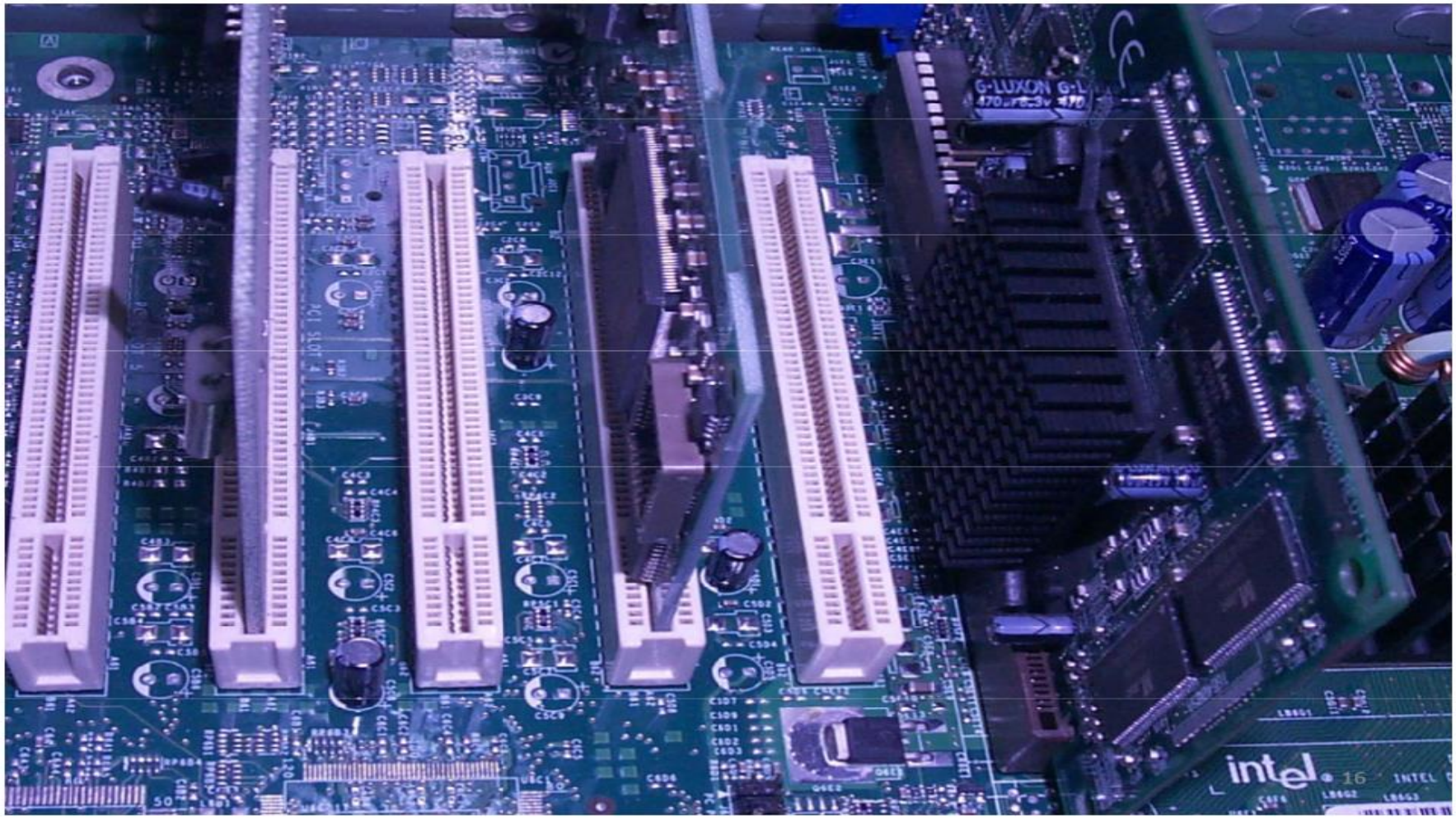
Hardware Abstraction



Hardware



Hardware



Hardware

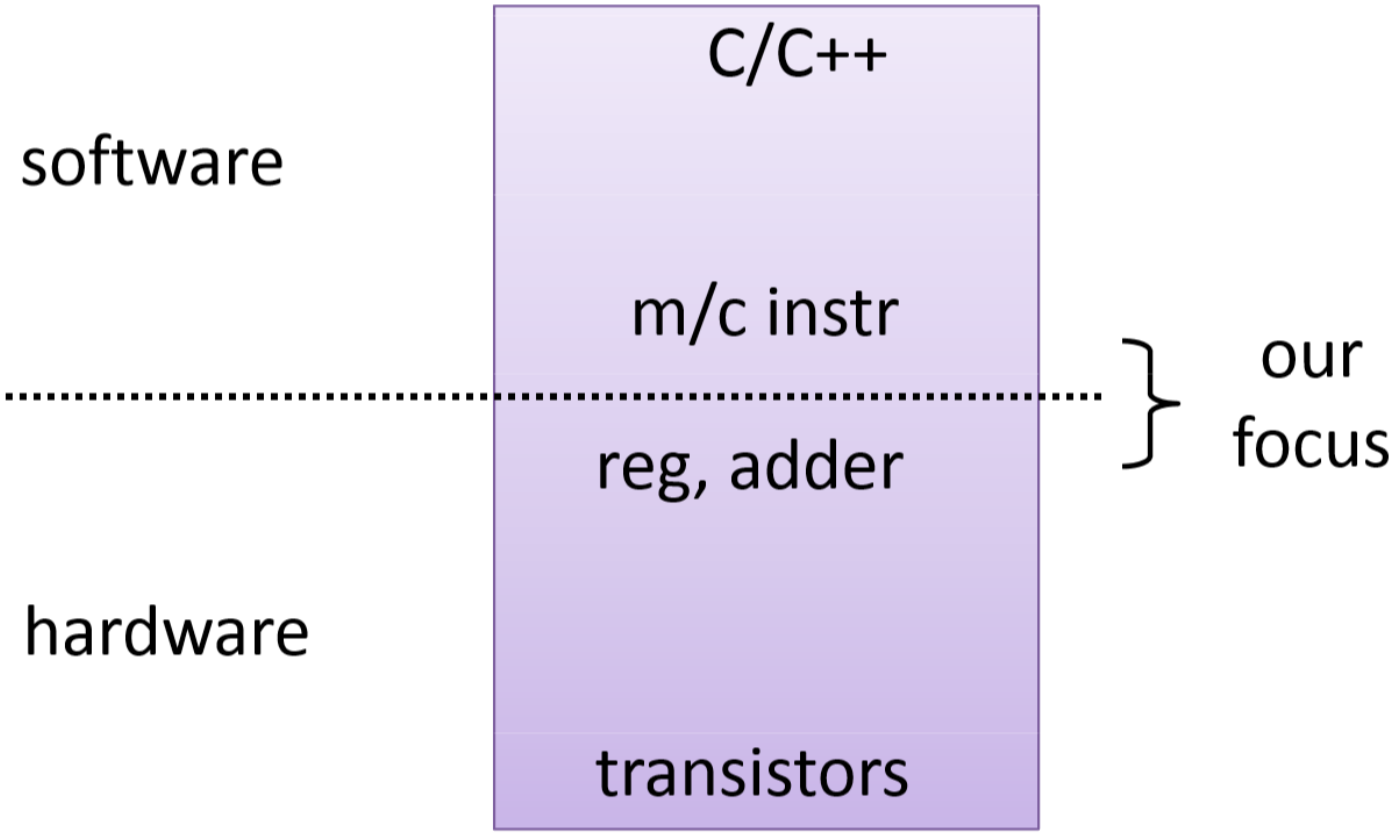
Hardware


Seagate
WARNING
MODEL
ST3290A
915007-608
CBSHBLD16
07-5.04-A1
Seagate
Barracuda ATA II
Model ST320420A
20.4 Gbytes
BARRACUDA
ULTRA
FRAGILE
DRIVE PARAMETERS

Hardware


22840
Seagate
Barracuda ATA II
Model ST320420A
20.4 Gbytes
FRAGILE

Hardware/Software Interface

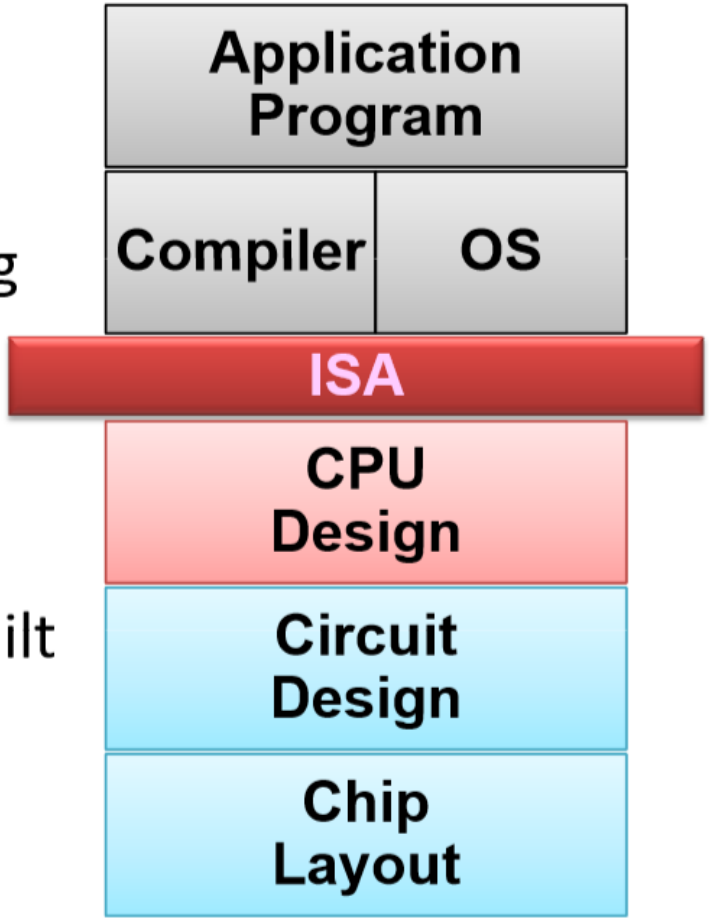


Architecture Levels

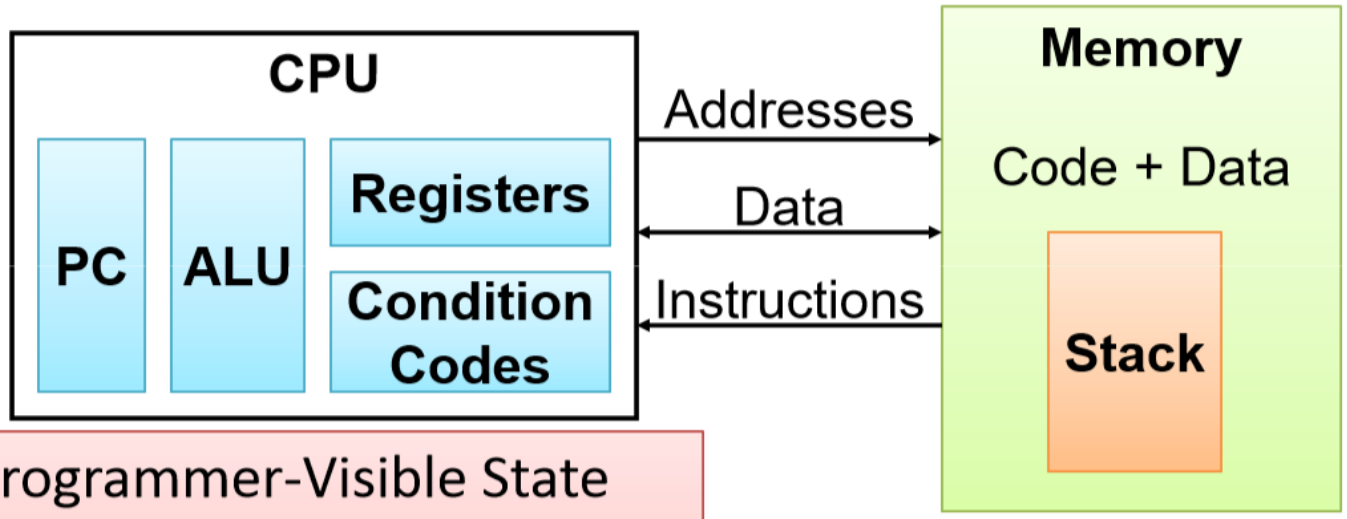
- Instruction set architecture (ISA)
  - Lowest level visible to a programmer
  - Operation (add/sub/mul/shift)
- Micro architecture
  - Fills the gap between instructions and logic modules
  - Operation Vs Micro Operation

Instruction Set Architecture

- Assembly Language View
  - Processor state (RF, mem)
  - Instruction set and encoding
- Layer of Abstraction
  - Above: how to program machine - HLL, OS
  - Below: what needs to be built - tricks to make it run fast



The Abstract Machine



- Programmer-Visible State
  - PC Program Counter
  - Register File
    - Heavily used data
  - Condition Codes

- Memory
  - Byte array
  - Code + data
  - stack