CS343: Operating System

Introduction to Operating System

Lect01 : 28th July 2023

Dr. A. Sahu

Dept of Comp. Sc. & Engg.

Indian Institute of Technology Guwahati

Outline

- Course, Attendance, Reference Book
- What do we study in this course?
- Why should this be studied?
- What is *"Operating System"* ?
- How is the course structured?

CS433 : Course site, Venue & Timing

- Course website: **http://jatinga.iitg.ernet.in/~asahu/cs343/**
- Class Venue : Core 5, Room- 5G4
- Class Timing & Venue
  - Mon 3.00PM-4.00PM
  - Tue 2.00PM-3.00PM
  - Fri 4.00PM-5.00PM
  - ***Thu 5.00 PM-6.00PM (Slot for makeup classes and Quizs)***

Course Pre requisite

- **Require Knowledge of**
  - CS204 : Data Structure and Algorithms
    - Use of Algorithm design, Graph, Analysis, Approximation, Heuristics, etc..
  - CS223 : Computer Org. & Architecture
    - Interface to BIOS, Architecture and ISA

CS343 OS: Text & Ref Books

- Text Book
  - Silberschatz, A. and Galvin P. B, ***Operating System Concepts***, 9/e. Wiley, 2018
- Reference Book
  - Stalling, W. ***Operating Systems: Internals and Design Principles*** , 6/e. Pearson, 2008.
  - Tanenbaum, A. S ***Modern Operating System***. 3/e. Pearson, 2007.
  - Dhamdhere, D. M ***Operating Systems: A Concept Based Approac***h, McGrawHill, 2008.

CS343 OS: Other Ref. Books

- Others Reference Book
  - Maurice Herlihy, Nir Shavit, ***Art of Multiprocessor Programming***, Elsevier 2009
  - C. Crowley ***Operating Systems: A Design-Oriented Approach***, Tata McGraw - Hill Education, 2009
  - Buttazzo Giorgio C. ***Hard Real-Time Computing Systems*** , Springer Verlag, 2011
  - P. Brucker ***Scheduling Algorithms*** , Springer-verlag, 2007
  - J. Corbet, A. Rubini ***Linux Device Drivers***,,3rd Ed. O'Reilly Media, 2005

CS343 : Grading and Rules

- **75% Attendance is Mandatory**
  - Attendance <75% : you are not to appear in End Semester Examination
- Grading
  - ***5% class participation***
  - 35% mid semester + 40% end semester
  - 20% Quiz
    - 2 Quiz before Mid Semester
    - 2 Quiz after Mid Semester

Course Structure: *1st Half*

- Process Management
  - Process and thread, scheduling examples
  - ***Scheduling Algorithms: Theoretical prospects***
- Concurrency
  - Mutual exclusion, synch., semaphores, deadlocks
  - Atomic Instructions, ***design and proof of Synchronization algorithms and polices***
- Memory Management
  - Allocation, protection, hardware support, paging, segmentation, virtual memory, demand paging, allocation, replacement, TLBs
  - ***Algorithmic treatment: Memory management***

Course Structure: *2nd Half*

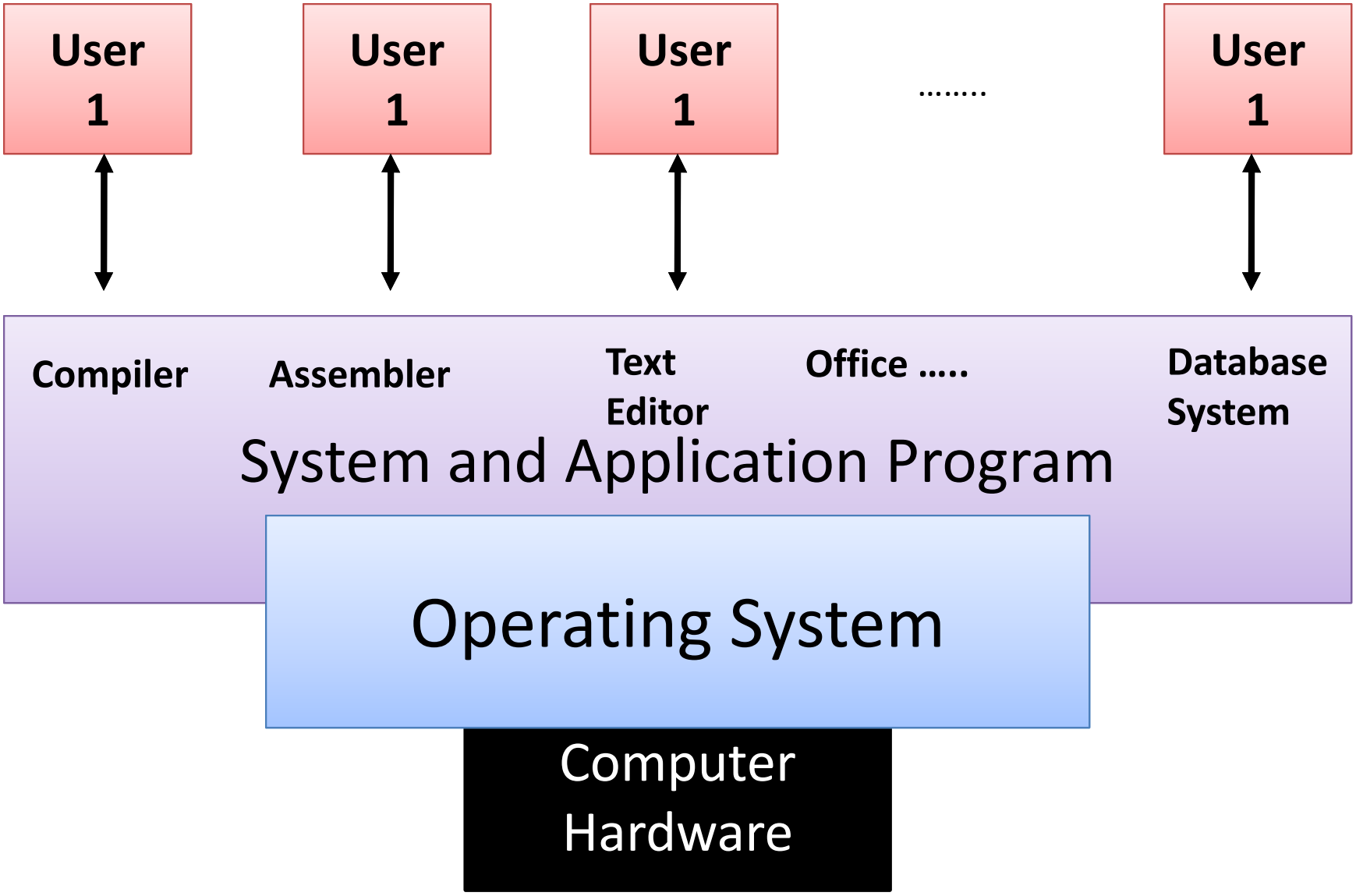
- File Management and File Systems
  - Naming, file operations and their implementation;
  - Allocation, free space management, directory management, mounting;
  - ***Distributed File System***
- I/O Management
  - Device drivers, disk scheduling
  - **Linux Device Driver & Kernel Programming**

Introduction
to
Computer System

Computer System Structure: Four Components

- Hardware : Provides basic computing resources
  - CPU, memory, I/O devices
- Operating system:
  - Controls and coordinates use of hardware among various applications and users
- Appl^n programs
  - Define the ways in which the system resources are used to solve the computing problems of the users
  - Word processors, ***compilers, assembler,*** web browsers, database systems, video games
- Users : People, machines, other computers

Four Components of a Computer System



Introduction to Computer System

- A computer system consists of
  - Hardware, **System programs**, Appl^{n} programs, Users

User

Office Word, Excel | **Banking System** | **Web Browser**

Application Program

Compilers | Editor | Command Interpreter

Operating System

System Program

Machine Language

Micro-architecture

Physical Devices

Hardware

OS : Laymen Definition

An operating system manages all:

- **Input**
  - getting information into the computer from an external sources
  - keyboard, a mouse, a scanner, or a disk.
- **Processing**
  - After receiving input: manipulates or alters the data
- **Output**
  - Once the input has been processed
  - Result output to a monitor, printer, disk or sent via email or the Web

OS Examples

- **PC/Laptop OS**
  - Microsoft : Window XP, Vista, 8, 10, 11, NT
  - Apple : Machintos, IBM : OS 2, OS 360/390
  - Unix, Linux, Ubuntu, Fedora, BSD Unix, Solaris
- **Embedded OS**
  - Android, iOS, Window CE/Phone 8.1, Bada OS, QNX, MeeGo, BlackBarry, uLinux, TinyOS
- **Web Browser OS :** Chrome OS, EyeOS, YouOS
- **Router OS:** CSIR ONET, Netware, Cisco IOS, SAN-OS
- TV OS : Samsung Tizen, TvOS (from Apple), Roku, LG WebOS, GoogleTV, FireTV, Android TV

OS Types

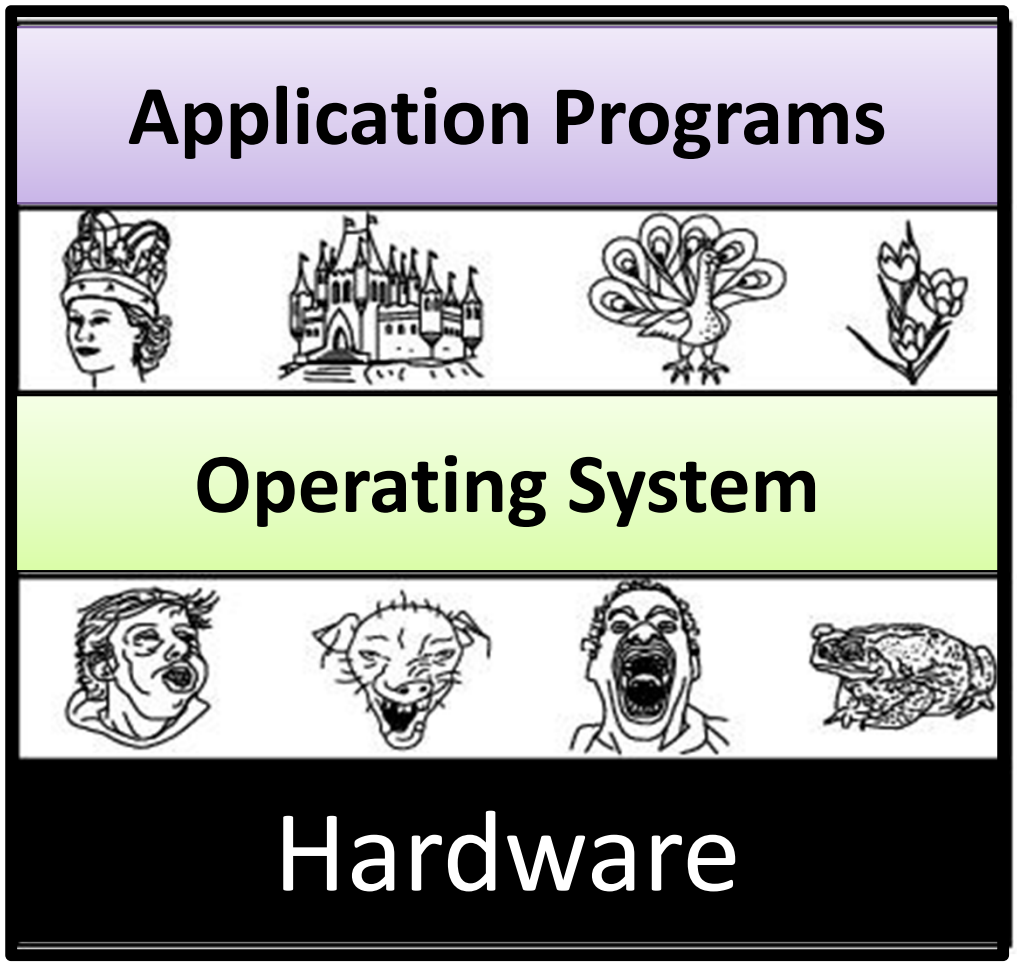
- Mainframe OS, Server OS
- Multiprocessor operating systems
- Personal computer operating systems
- Real-time operating systems
  - Air craft, Radar Detection, Naval/Space Machine
- Embedded operating systems
  - Mobile, Printer, Scanner, Projector, Camera, Washing Machine
- Smart card operating systems
  - Ecos, TinyOS, SensorOS

What is an Operating System

- **OS is an extended machine**
  - Hides the messy details which must be performed
  - Presents user with a virtual machine, easier to use
- **OS is a resource manager**
  - Each program gets time with the resource
  - Each program gets space on the resource

OS as an Extended Machine

Operating systems turn ugly hardware into beautiful abstractions.



Beautiful Interface

Ugly Interface

OS as a Resource Manager

- Allow multiple programs to run at the same time
- Manage and protect memory, I/O devices, and other resources
- Includes multiplexing (sharing) resources in two different ways:
  - In time
  - In space

Abstraction

- Delving into the depths reveals more information

- An abstraction omits unneeded detail, helps us cope with complexity

Software Abstraction

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

C

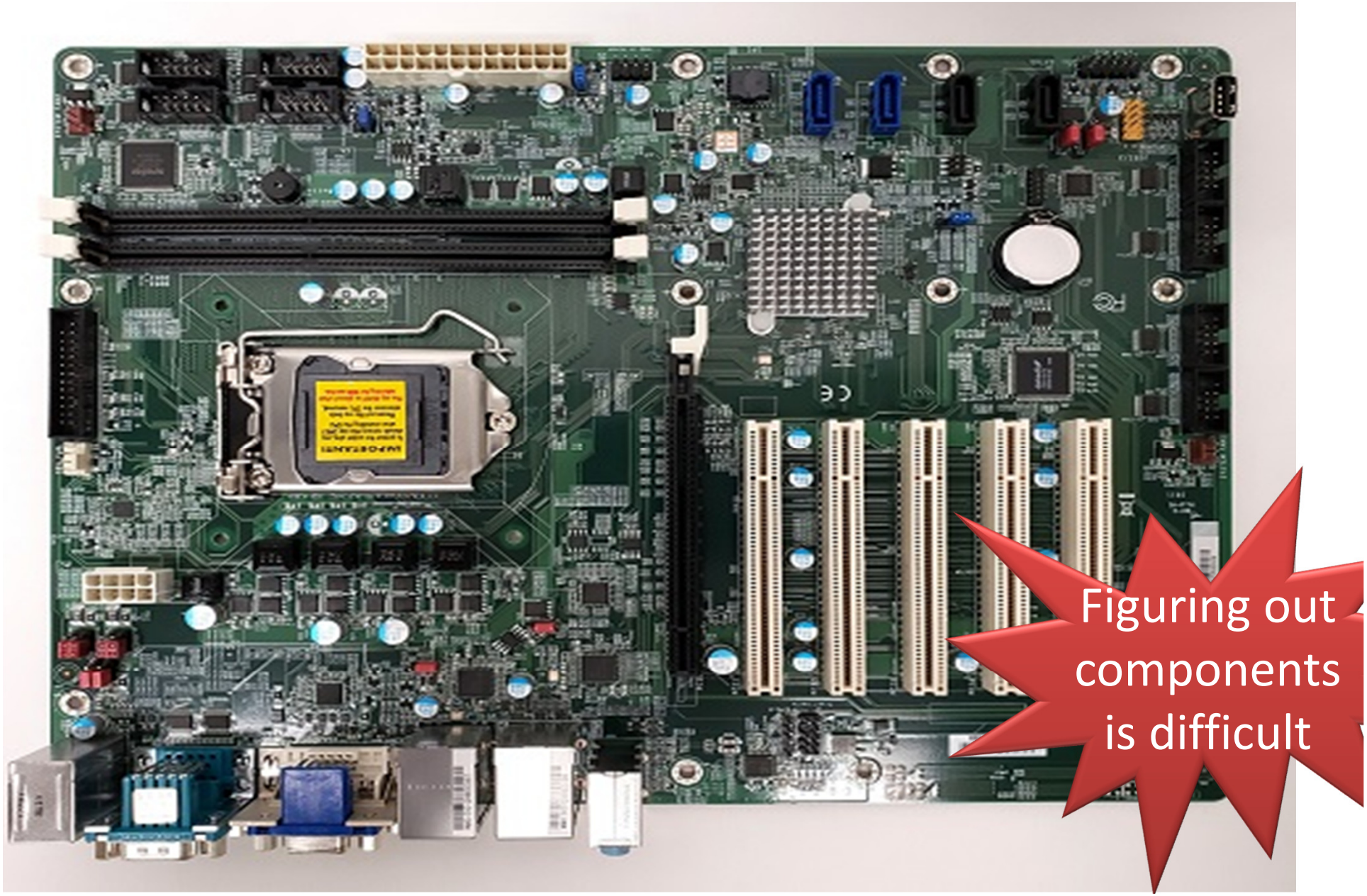
```
_sum:
      pushl %ebp
      movl %esp,%ebp
      movl 12(%ebp),%eax
      addl 8(%ebp),%eax
      movl %ebp,%esp
      popl %ebp
      ret
```

assembly

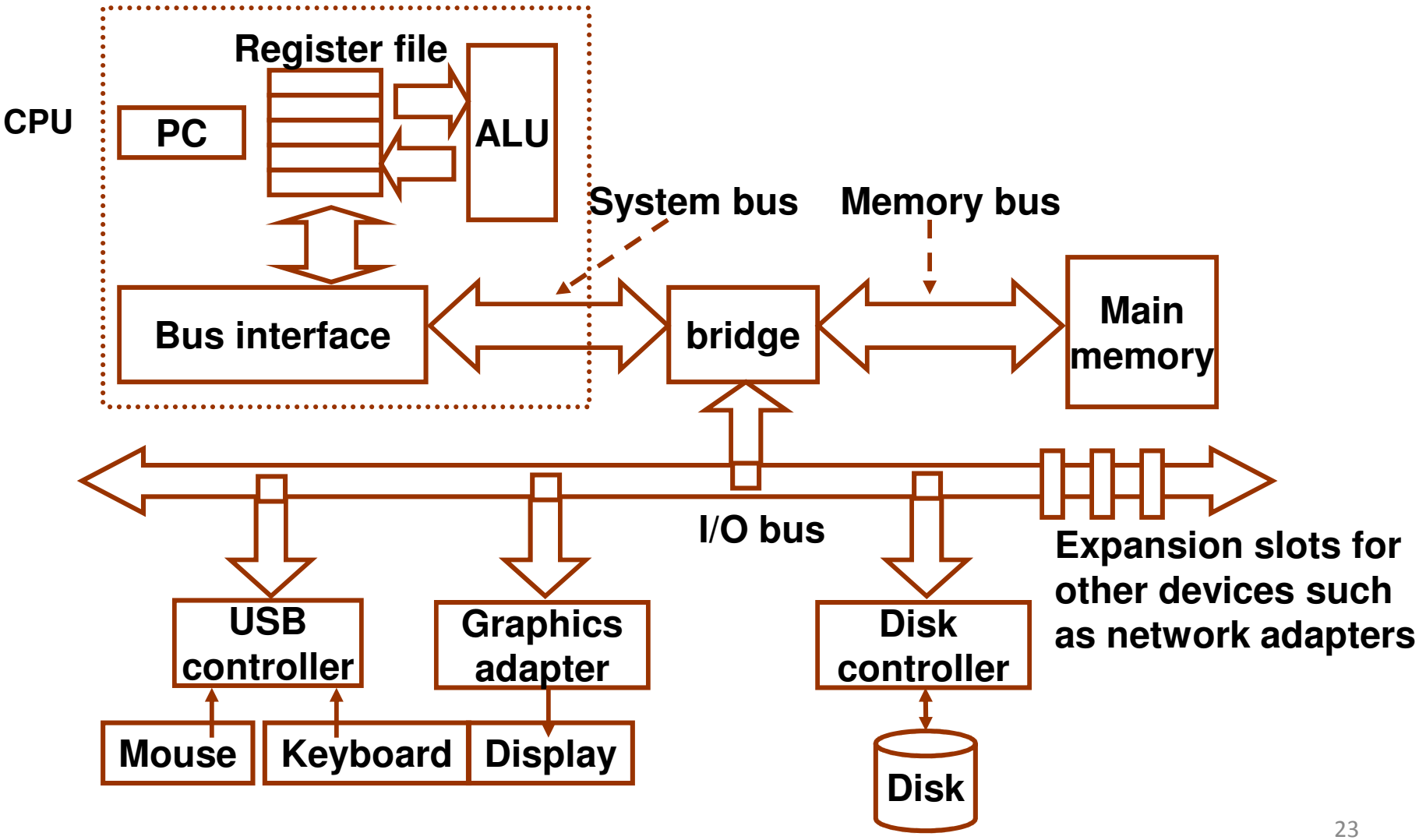
```
0x401040 <sum>:      0x55
                     0x89
                     0xe5
                     0x8b
                     0x45
                     0x0c
                     0x03
                     0x45
                     0x08
                     0x89
                     0xec
          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

machine code

Example: Hardware Abstraction



Hardware Abstraction



Thanks