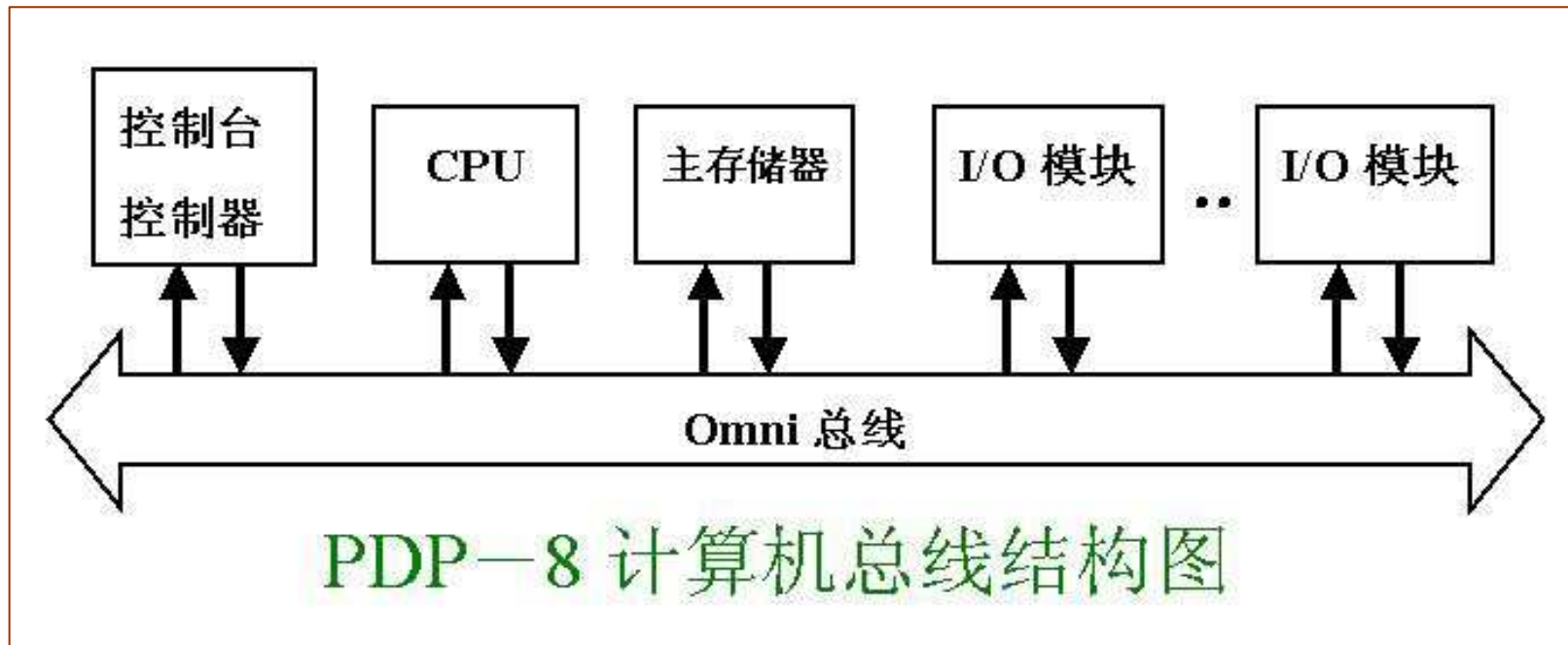

Lecture 2: Computer Abstractions II

DEC公司的PDP-8机

- 同在**64**年出现。与**IBM 360**相比，价格更低、更小巧，因而被称为小型机（**Minicomputer**）
- **PDP-8**“创造了小型机的概念，并使之成为数十亿美元的工业”，使**DEC**成为了最大的小型机制造商。
- 主要特点：首次采用总线结构。

Omnibus总线包含了**96**个独立的信号通道，用以传送控制、地址和数据信号。这种结构具有高度的灵活性，允许将模块插入总线以形成各种配置。

PDP-8/E计算机系统框图



问题：“总线结构”有什么好处？

具有高度的灵活性，允许将模块插入总线以形成各种配置
节省器件，体积小，价格便宜

计算机发展简史

◦ (第四代：LSI/VLSI/ULSI 1972~至今)

- 微处理器和半导体存储器技术发展迅猛，微型计算机出现。使计算机以办公设备和个人电脑的方式走向普通用户。

半导体存储器

- 70年Fairchild公司生产出第一个相对大容量半导体存储器
- 74年位价格低于磁芯的半导体存储器出现，并快速下跌
- 从70年起，存储密度呈4倍提高（几乎是每3年）

微处理器

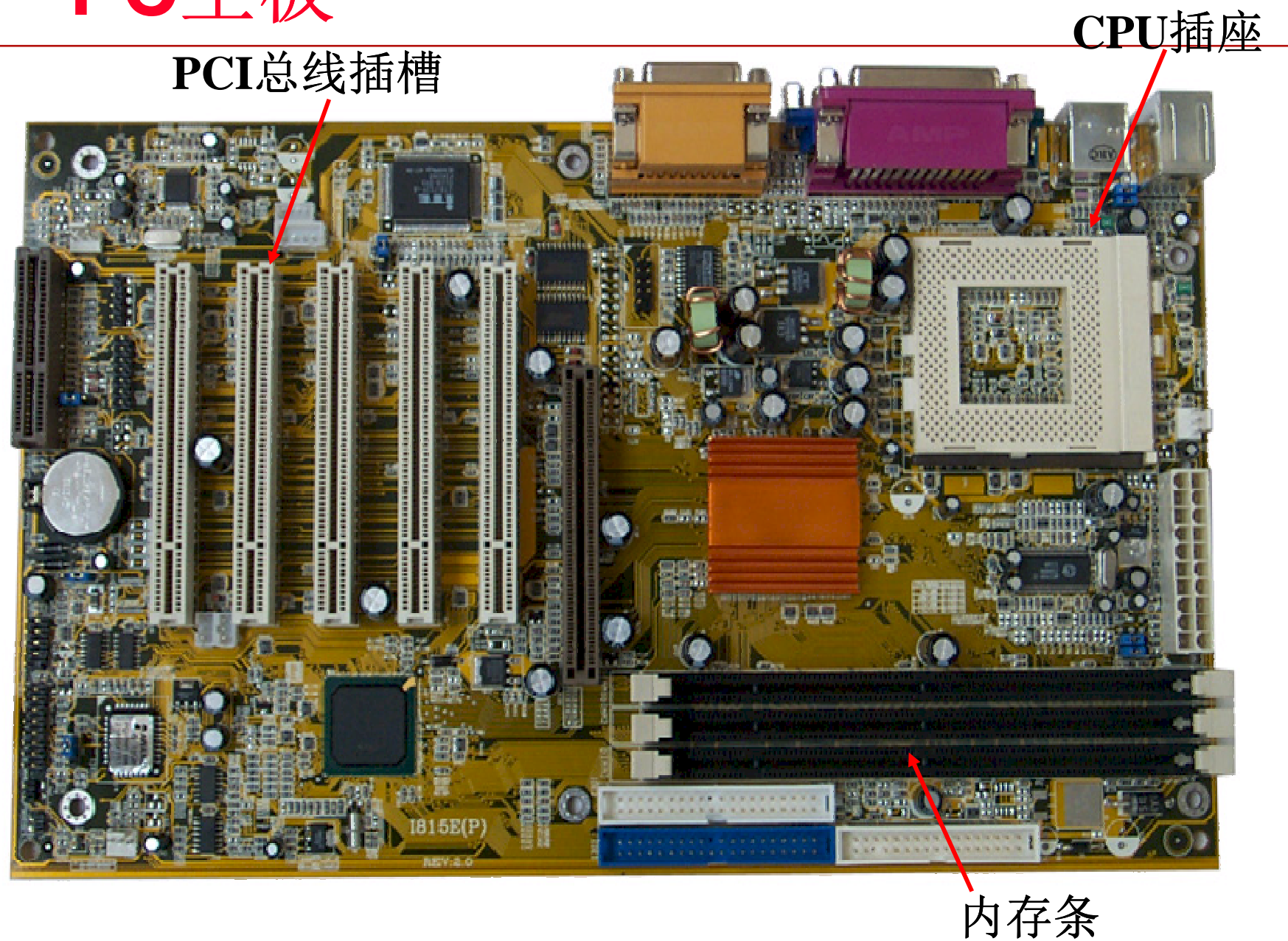
- 微处理器芯片密度不断增加，使CPU中所有元件放在一块芯片上成为可能。71年开发出第一个微处理器芯片4004。
- 特点：共享存储器，分布式存储器及大规模并行处理系统

以后几代（标准、意见不一）（注：有称第四代是VLSI，从80年代开始；也有称第四代是LSI，从72年开始；有的又分成LSI时代和VLSI时代）

计算机硬件：打开计算机来看看

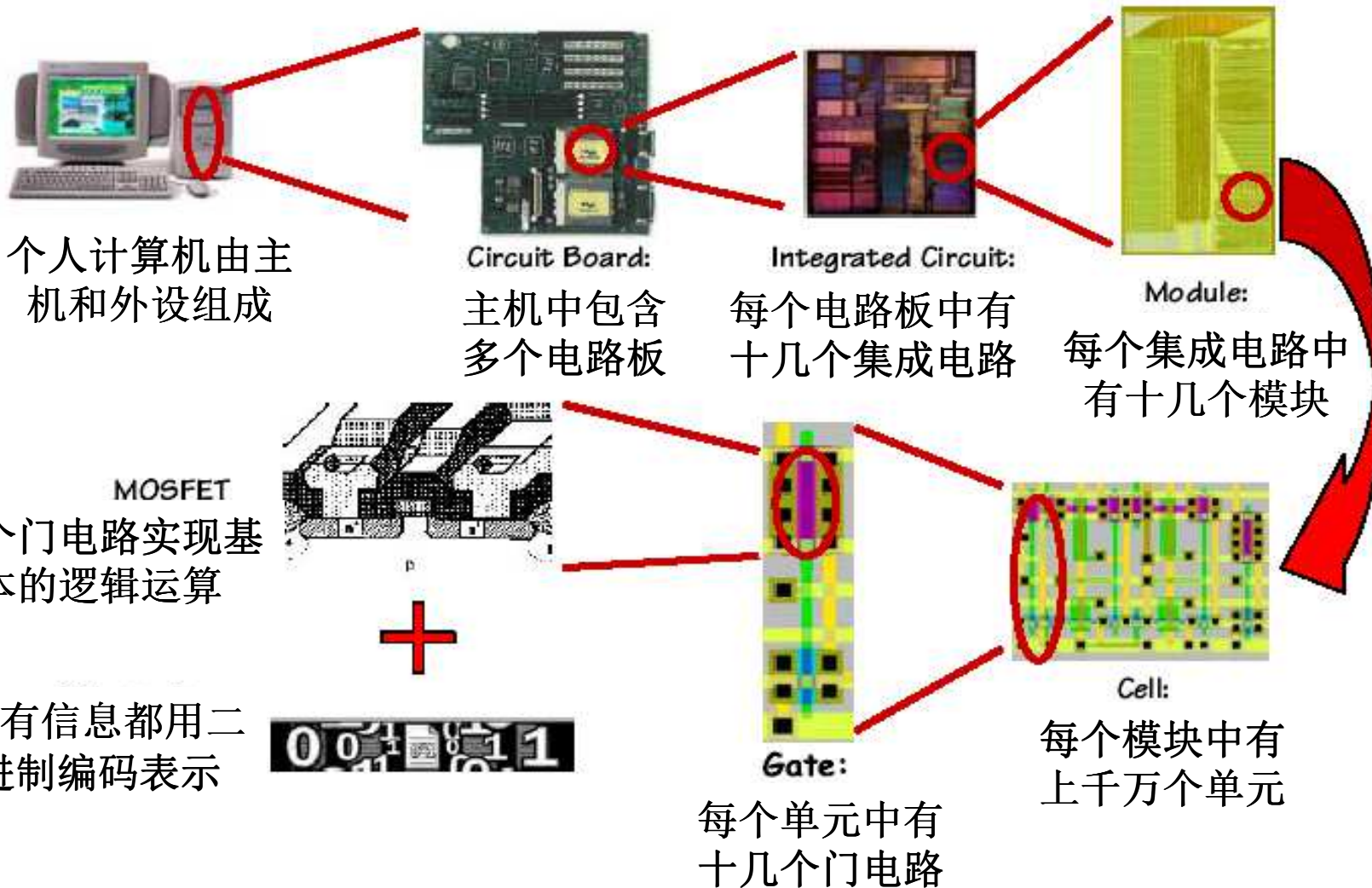


PC主板

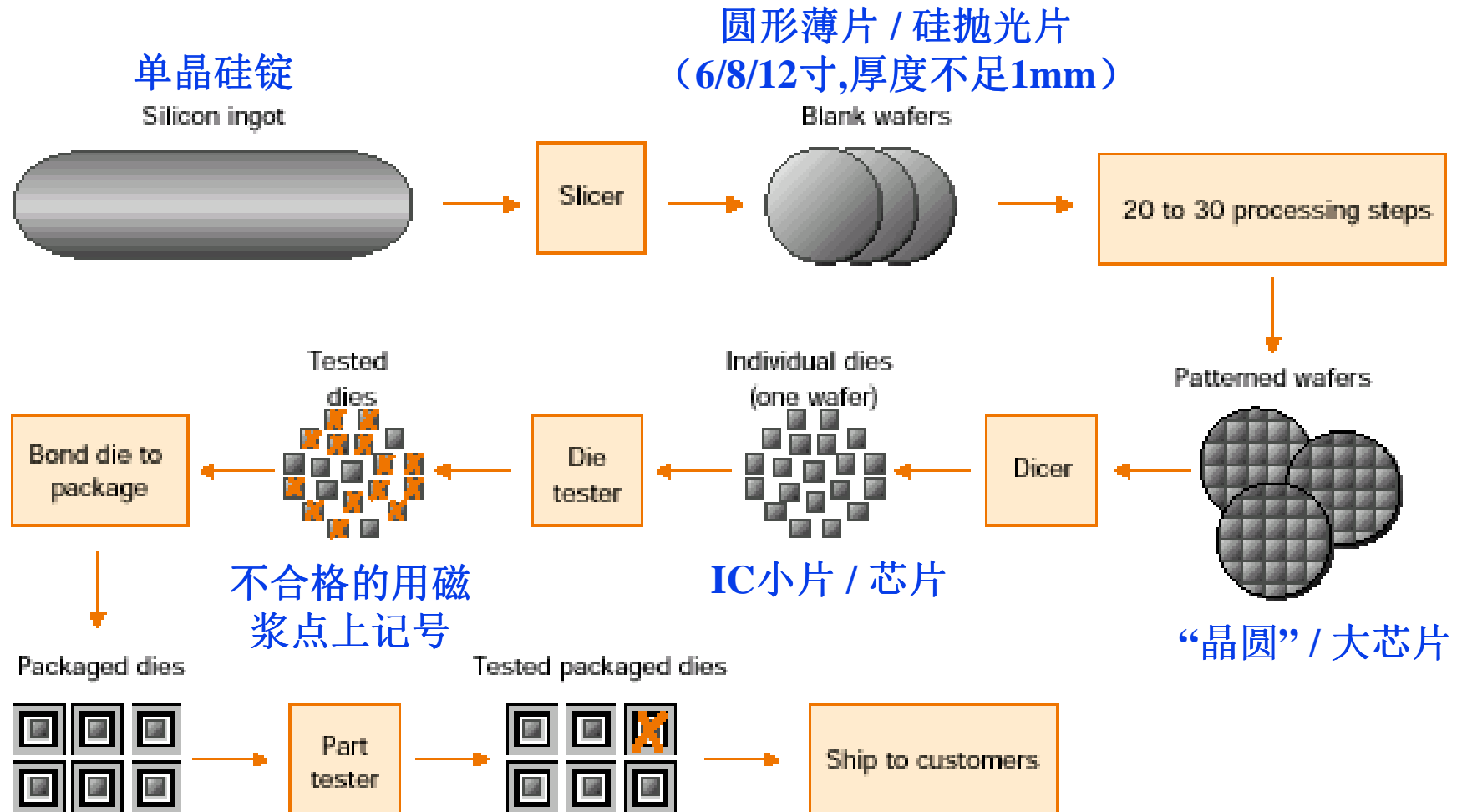


解剖一台计算机

How do you build systems with >1G components?



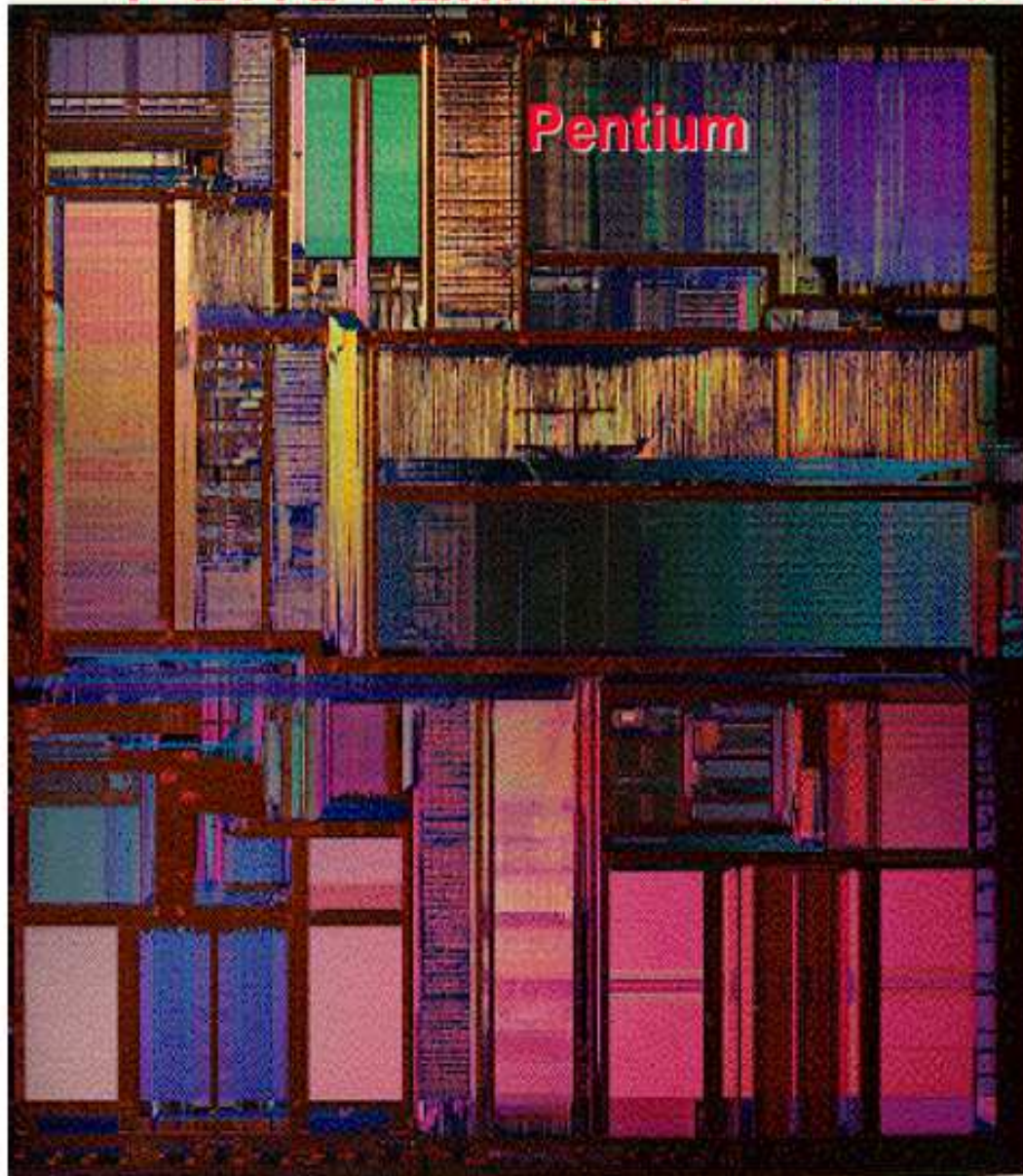
Integrated Circuits manufacturing process



封装：将芯片固定在塑胶或陶瓷基座上，把芯片上蚀刻出来的引线与基座底部伸出的引脚连接，盖上盖板并封焊成芯片

约需400多道工序！

Pentium芯片内的主要功能块



Die Area: 91 mm²

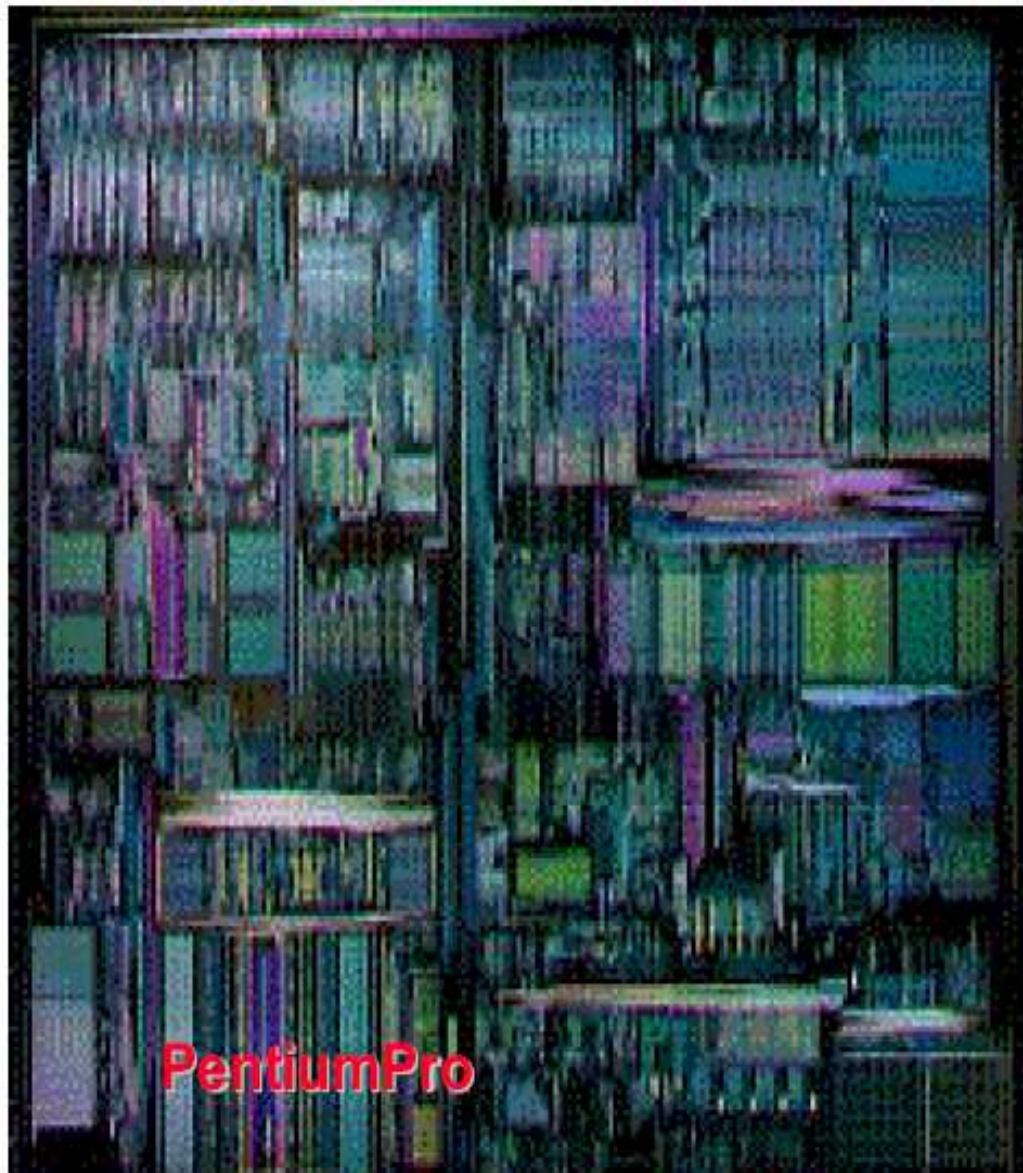
直径8 inch(200mm)的
Wafer最多可做 196个Die

≈ 3,300,000 Transistors

Cache: ≈1M Transistors

296 Pins

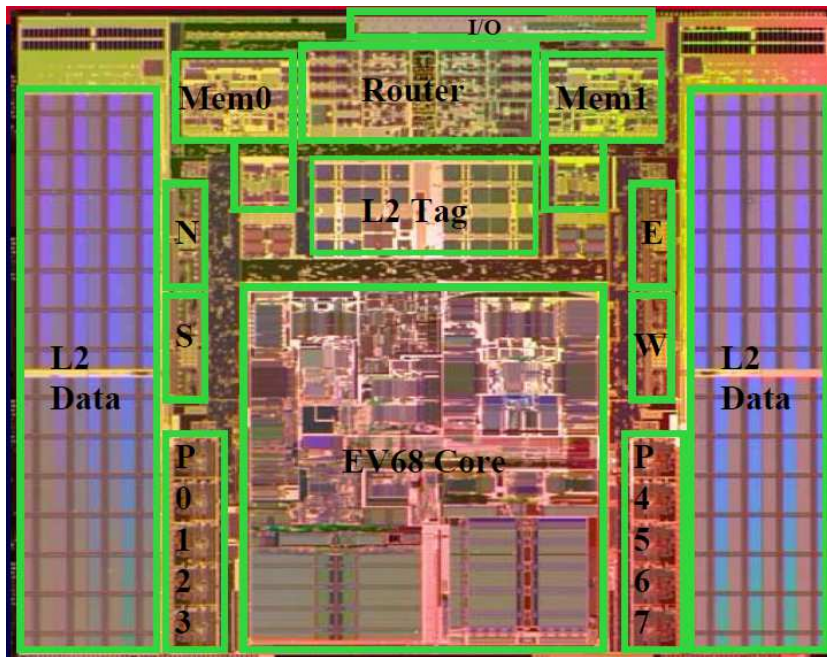
PentiumPro芯片内的主要功能块



- Die Area: 306 mm²
- 直径8 inch(200mm)的 Wafer最多可做 78个Die
- \approx 5,500,000 Transistors
- Cache: \approx 1M Transistors
- External Cache:
31M Transistors

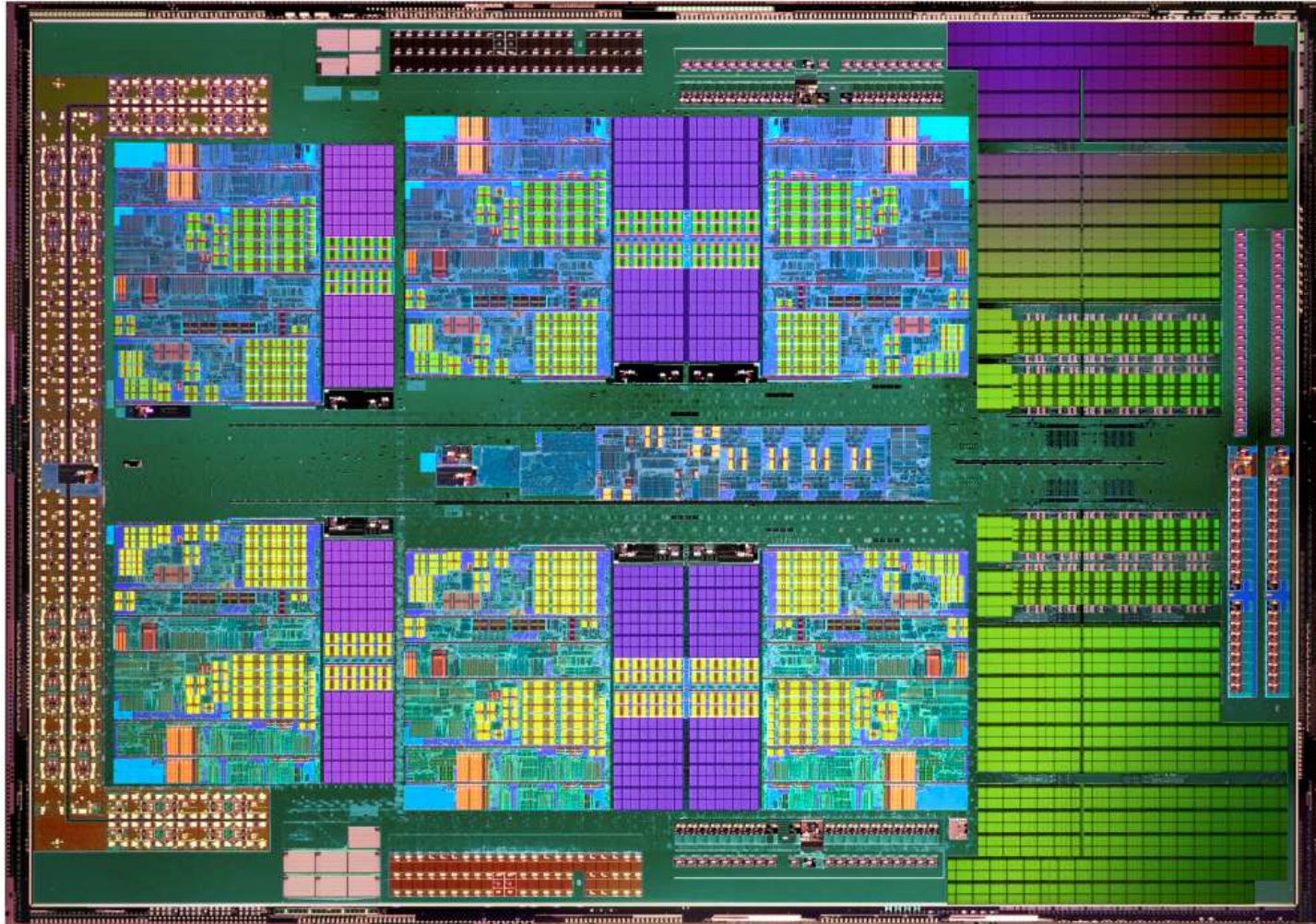
PentiumPro Package =
PentiumPro+ExternalCache
387 Pins

ALPHA 21364 Die Shot

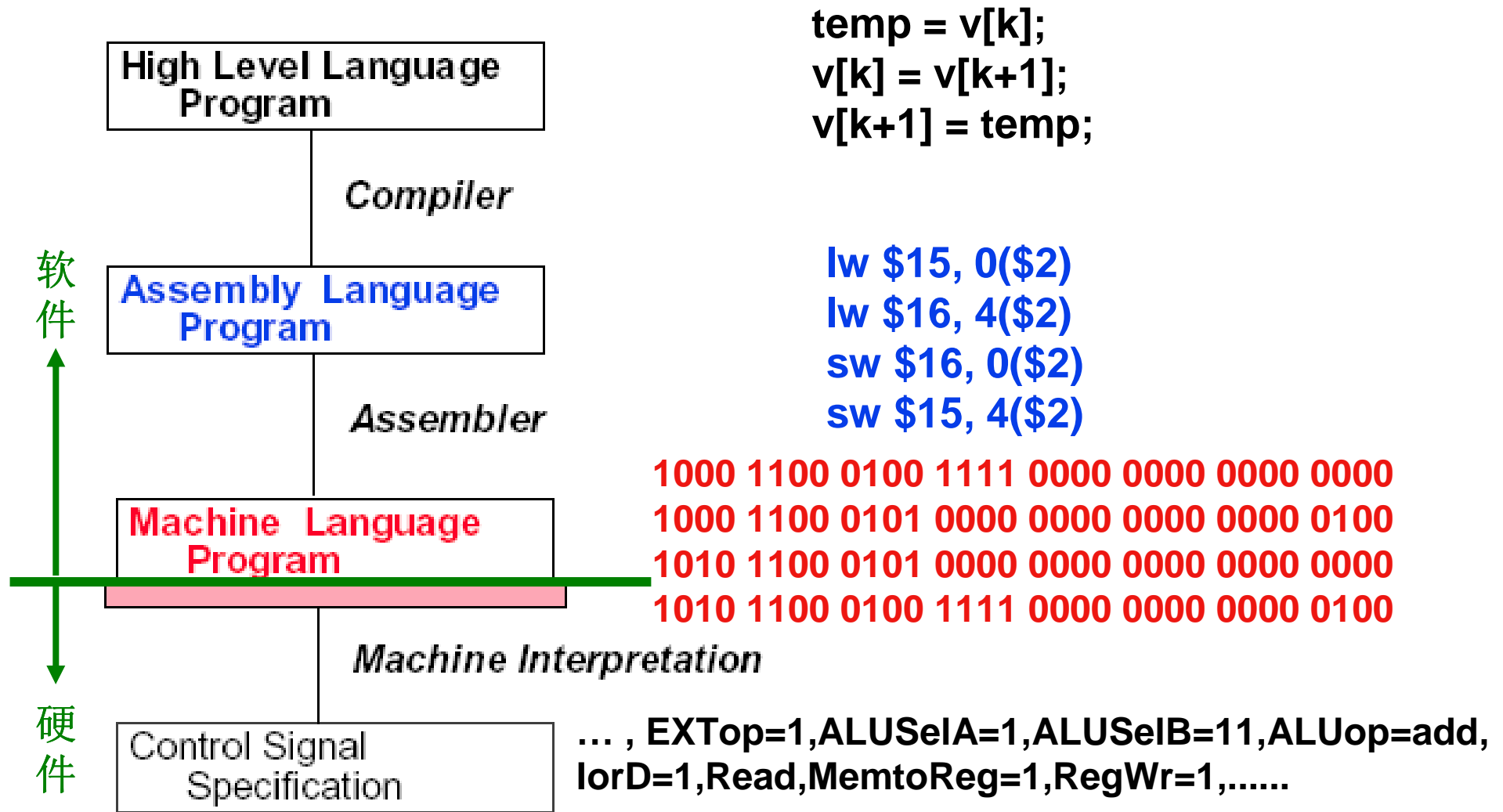


FPMul			IntReg
FPrep	IntMap	IntQ	
FAdd	FPQ	LdStQ	IntExec
BPred		ITB	
		DTB	
I-Cache		D-Cache	

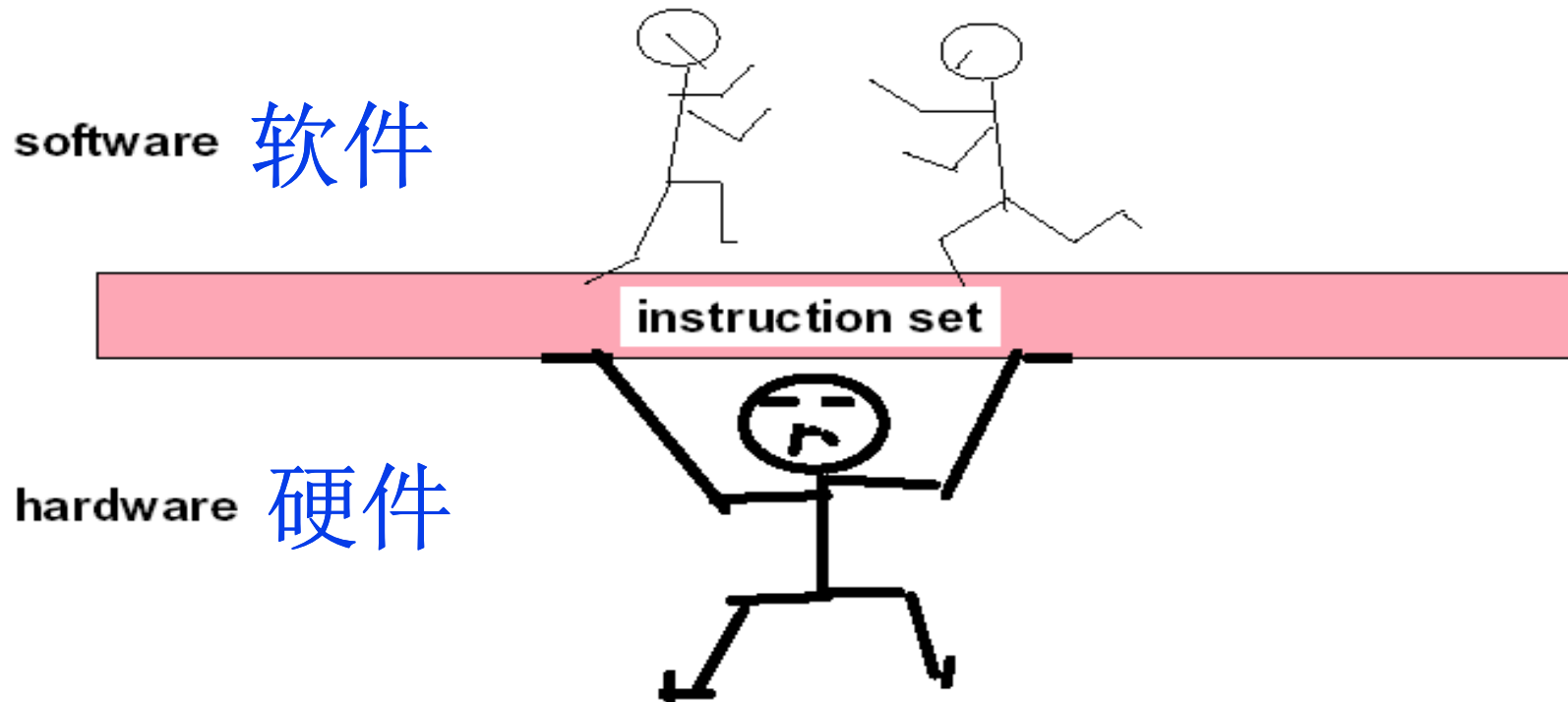
AMD Phenom II X6 Die Shot



Hardware/Software Interface



Hardware/Software Interface (界面)



软件和硬件的界面： **ISA** (Instruction Set Architecture)

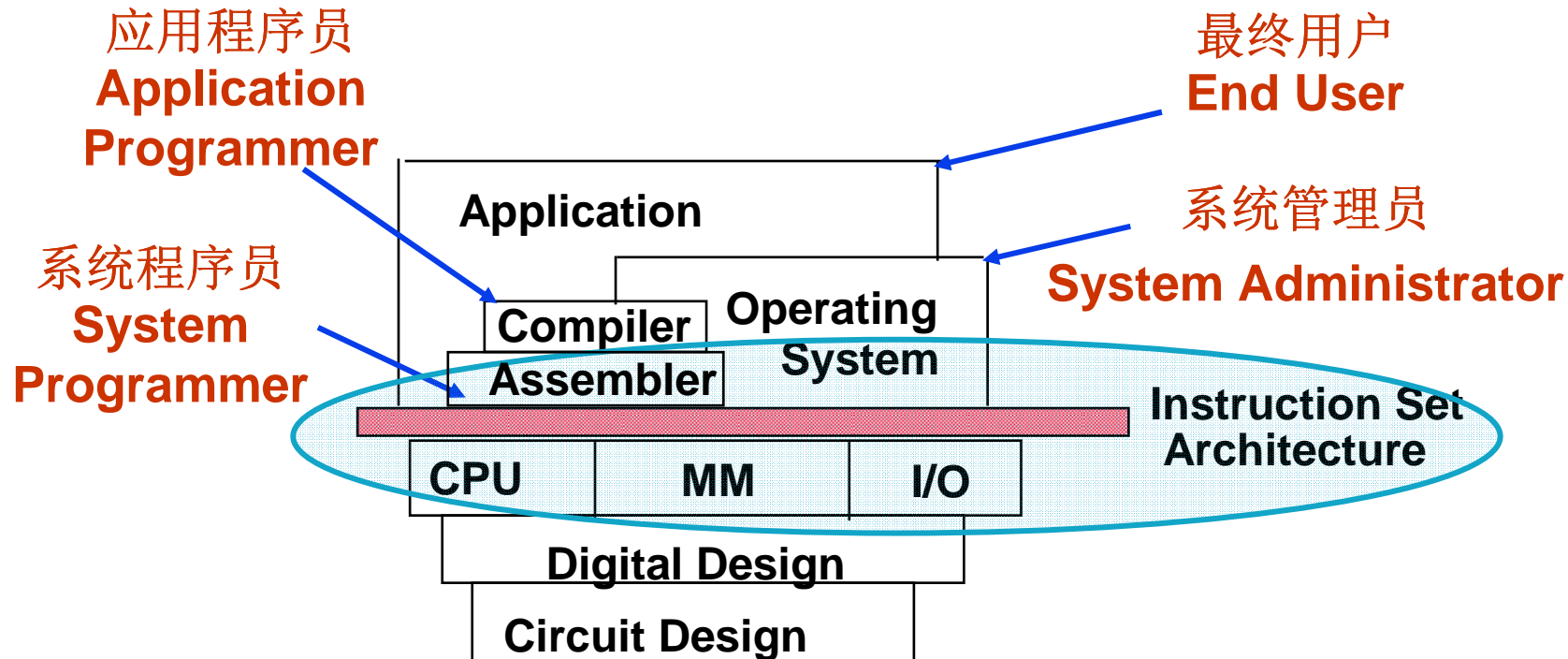
指令集体系结构

机器语言由指令代码构成，能被硬件直接执行。

Software

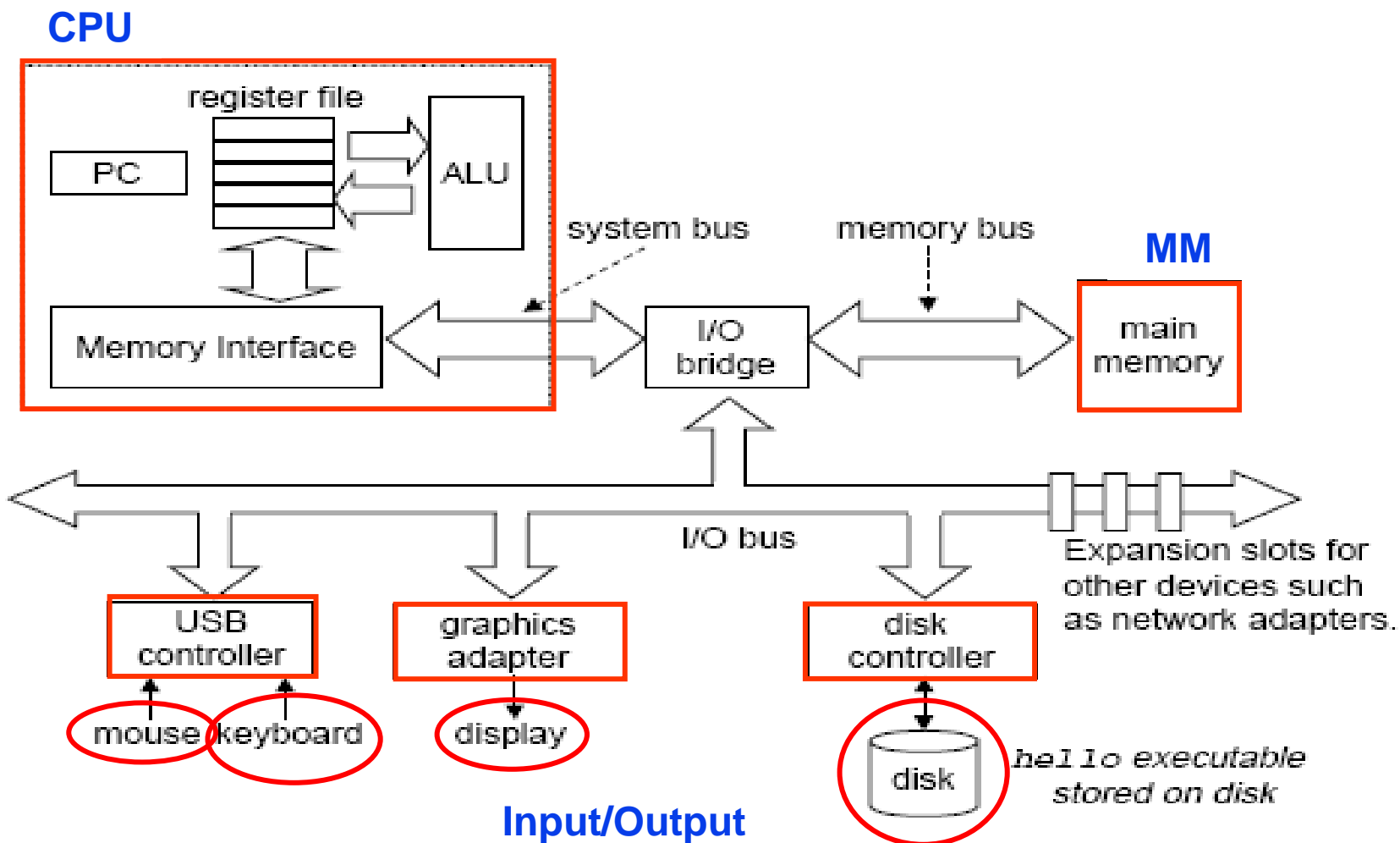
- **System software(系统软件)** - 简化编程过程，使硬件资源被有效利用
 - 操作系统 (**Operating System**) : 硬件资源管理，用户接口
 - 语言处理系统: 翻译程序+ **Linker, Debug, Loader, etc ...**
 - 翻译程序(**Translator**)有三类:
 - 汇编程序(**Assembler**): 汇编语言源程序→机器语言目标程序
 - 编译程序(**Complier**): 高级语言源程序→汇编/机器语言目标程序
 - 解释程序(**Interpreter**): 将高级语言语句逐条翻译成机器指令并立即执行,不生成目标文件。
 - 其他实用程序: 如: 磁盘碎片整理程序、备份程序等
- **Application software(应用软件)** - 解决具体应用问题/完成具体应用任务
 - 各类媒体处理程序: **Word/ Image/ Graphics/...**
 - 管理信息系统 (**MIS**)
 - **Game, ...**

Computer Hierarchy (计算机系统层次)



- 上图给出的是计算机系统的层次结构
指令系统（即**ISA**）是软/硬件的界面
- 不同用户工作在不同层次，所看到的计算机不一样
- 中间阴影部分就是本课程主要内容，处于最核心的部分！

一个典型系统的硬件组成



PC: 程序计数器; **ALU:** 算术/逻辑单元; **USB:** 通用串行总线

一个典型程序的转换处理过程

经典的“hello.c”C-源程序

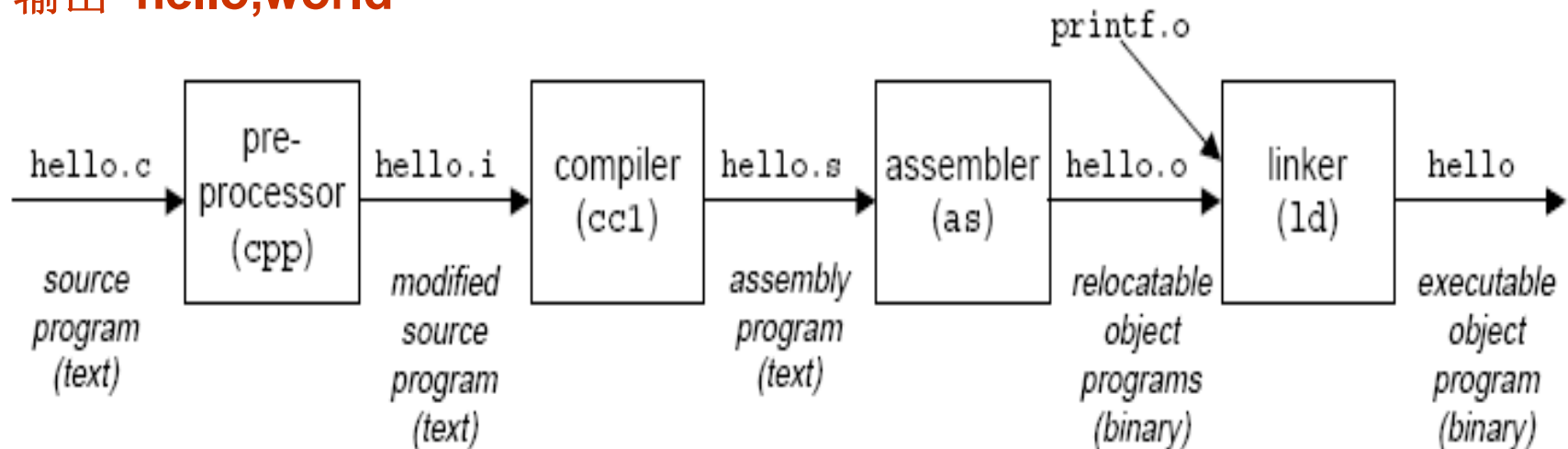
```
1 #include <stdio.h>
2
3 int main()
4 {
5 printf("hello, world\n");
6 }
```

程序的功能是：

输出“hello,world”

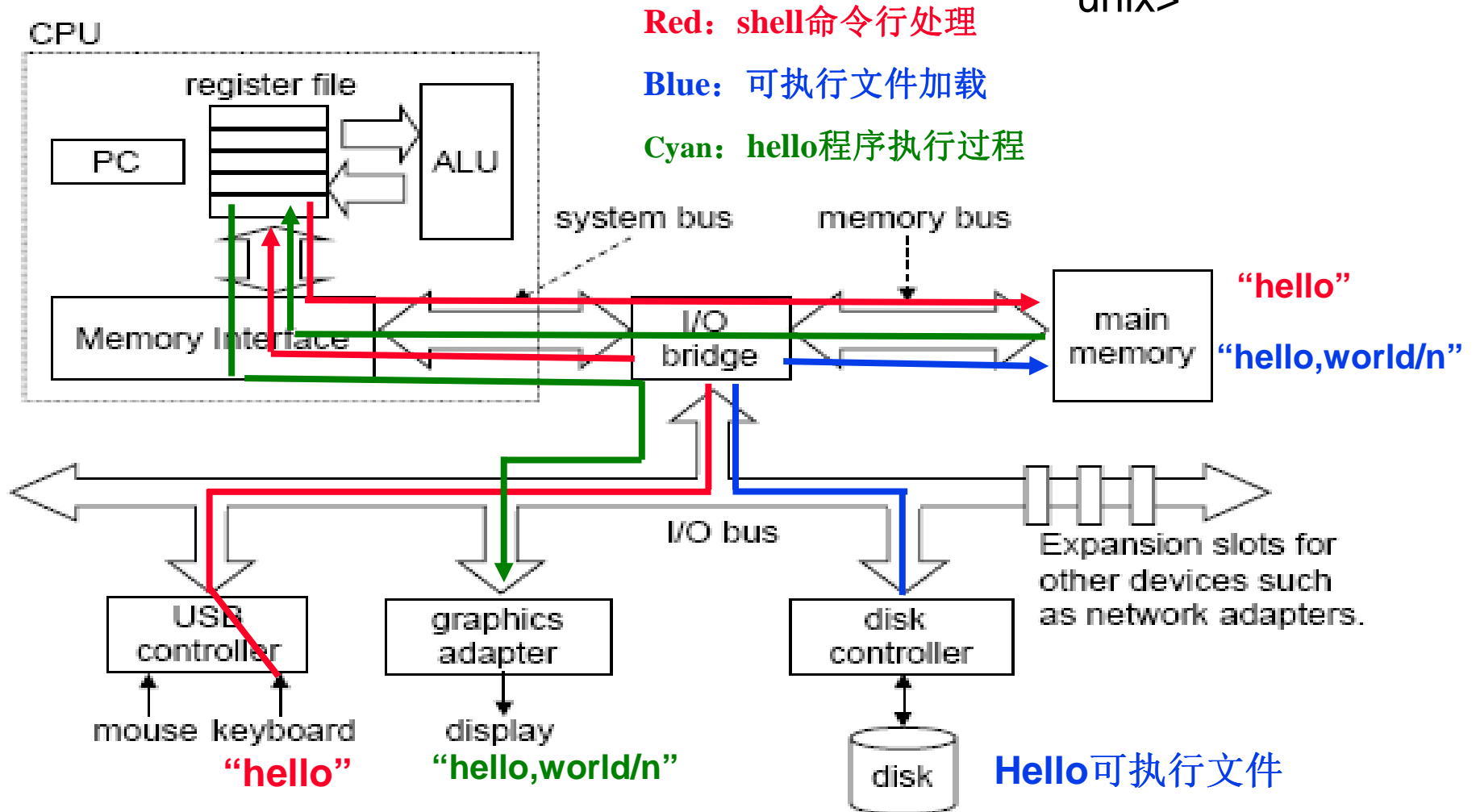
hello.c的ASCII文本表示

```
# i n c l u d e < s p > < s t d i o .
35 105 110 99 108 117 100 101 32 60 115 116 100 105 111 46
h > \n \n i n t < s p > m a i n ( ) \n {
104 62 10 10 105 110 116 32 109 97 105 110 40 41 10 123
\n < s p > < s p > < s p > < s p > p r i n t f ( " h e l
10 32 32 32 32 112 114 105 110 116 102 40 34 104 101 108
l o , < s p > w o r l d \n " ) ; \n }
108 111 44 32 119 111 114 108 100 92 110 34 41 59 10 125
```



Hello程序的数据流动过程

```
unix> ./hello [Enter]
hello, world
unix>
```

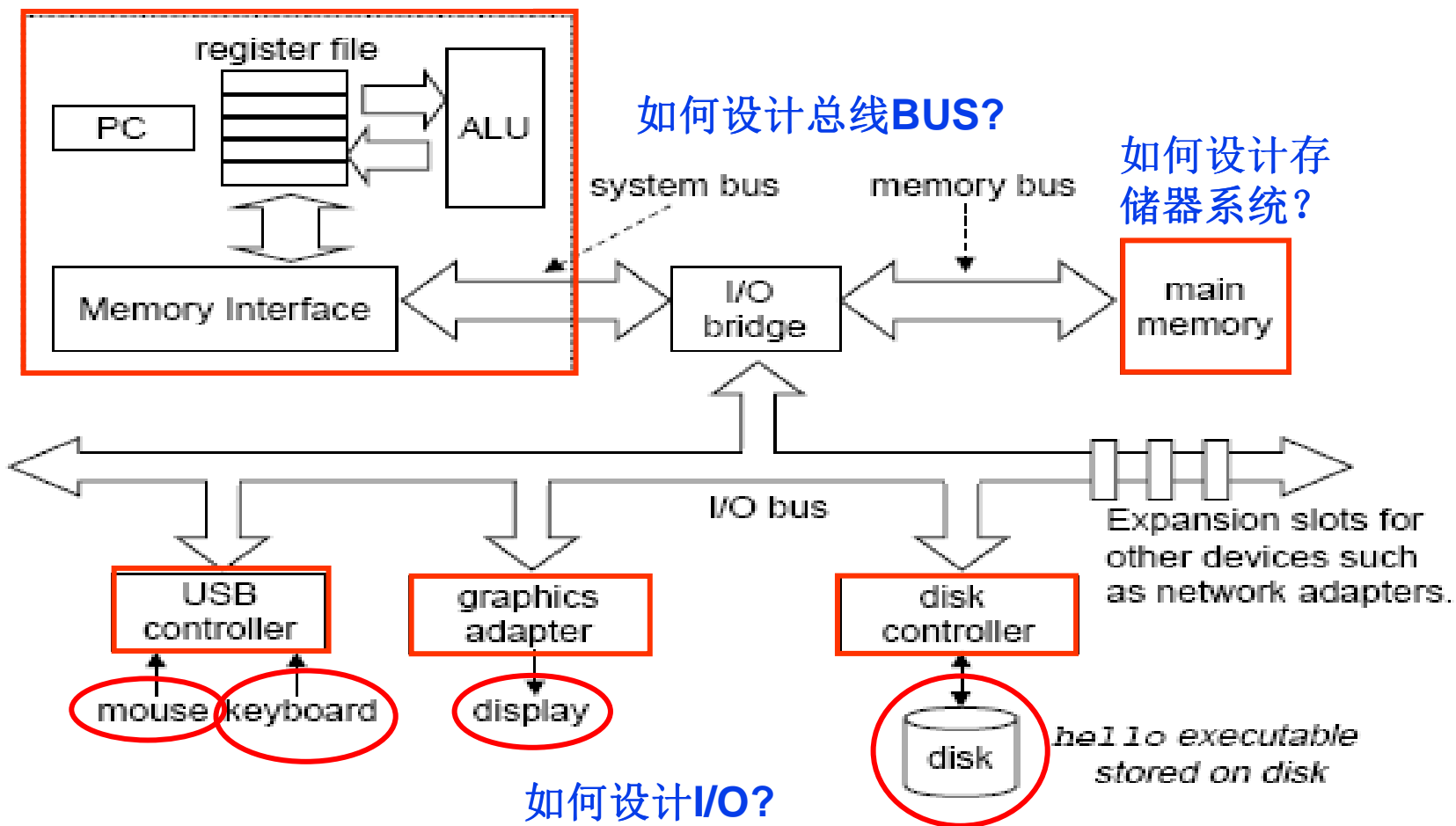


数据经常在各存储部件间传送。故现代计算机大多采用“缓存”技术！

所有过程都是在CPU执行指令所产生的控制信号的作用下进行的。

该课程的主要学习内容

如何设计高性能CPU?



信息（指令和数据）在计算机中如何表示？

指令系统如何设计？