



华北电力大学

North China Electric Power University

# 计算机组成与结构

华北电力大学计算机系

# 主要教学环节

- 课堂教学：56学时
- 综合实验（分散，16学时）
  - 运算器实验
  - 内存存储器实验
  - Flash存储器实验
  - CPU设计实验

# 学习目的

- 掌握计算机单机系统的硬件组成
  - 基本工作原理
  - 内部运行机制
  - 建立完整计算机系统概念
    - CPU整机概念
    - 硬件系统整机概念
- 培养计算机系统分析、设计和开发能力

# 主要教学内容

- 运算器的功能、组成和基本运行原理
- 存储器和层次存储系统原理
- 指令格式和指令系统
- 控制器的功能、组成和运行原理
- 输入/输出设备和总线
- 输入输出系统

# 学习方法

- 课堂讲授为主
- 阅读参考资料，自主学习和思考
- 通过实践环节，理解课程内容
- 思考、讨论和总结
- 复习考试

# 考试成绩

- 平时成绩
  - 课堂表现、课堂作业、课后作业
- 过程考试
  - 课堂测试+过程测试+小论文
- 结课考试
  - 闭卷考试
- 最终成绩=平时成绩+过程考试+结课考试

# 教材和参考资料

- 教材

- 计算机组成与结构（第二版），中国电力出版社

- 参考书目

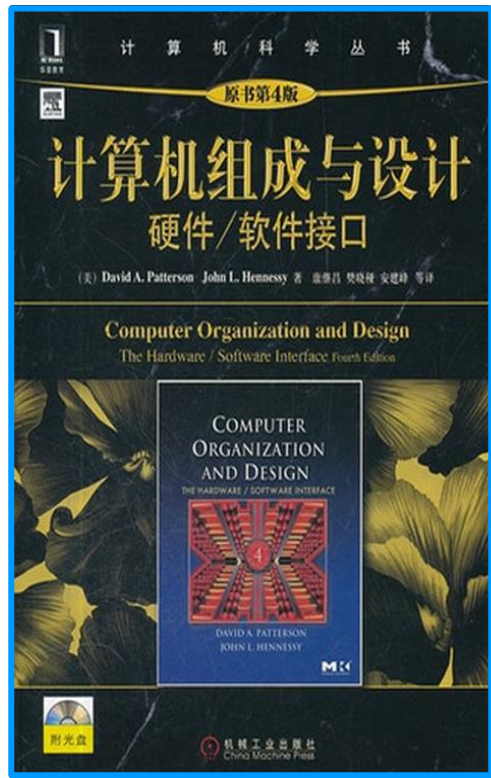
- 计算机组成原理，唐朔飞，高等教育出版社
  - 计算机组成原理，纪禄平，电子工业出版社
  - Computer Organization and Design  
The Hardware/Software Interface，机械工业出版社
  - Computer Systems – A Programmer’s Perspective，  
机械工业出版社

- 实验指导书

- 计算机硬件系统实验教程，清华大学出版社



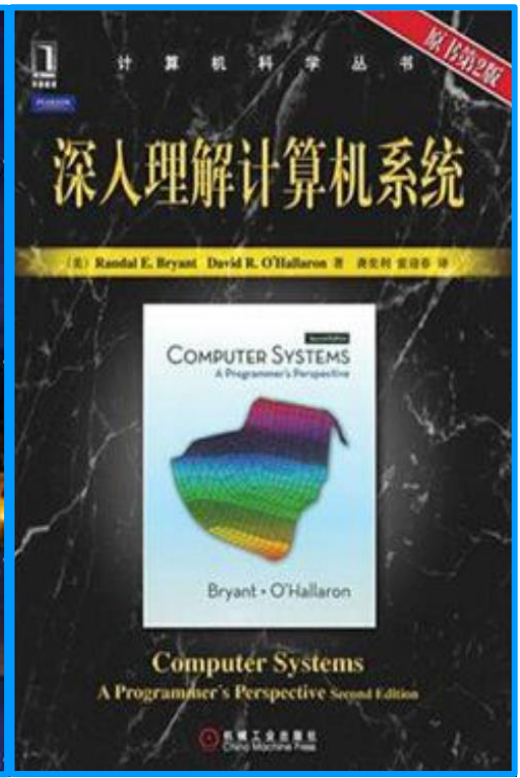
# 参考资料



Patterson & Hennessy

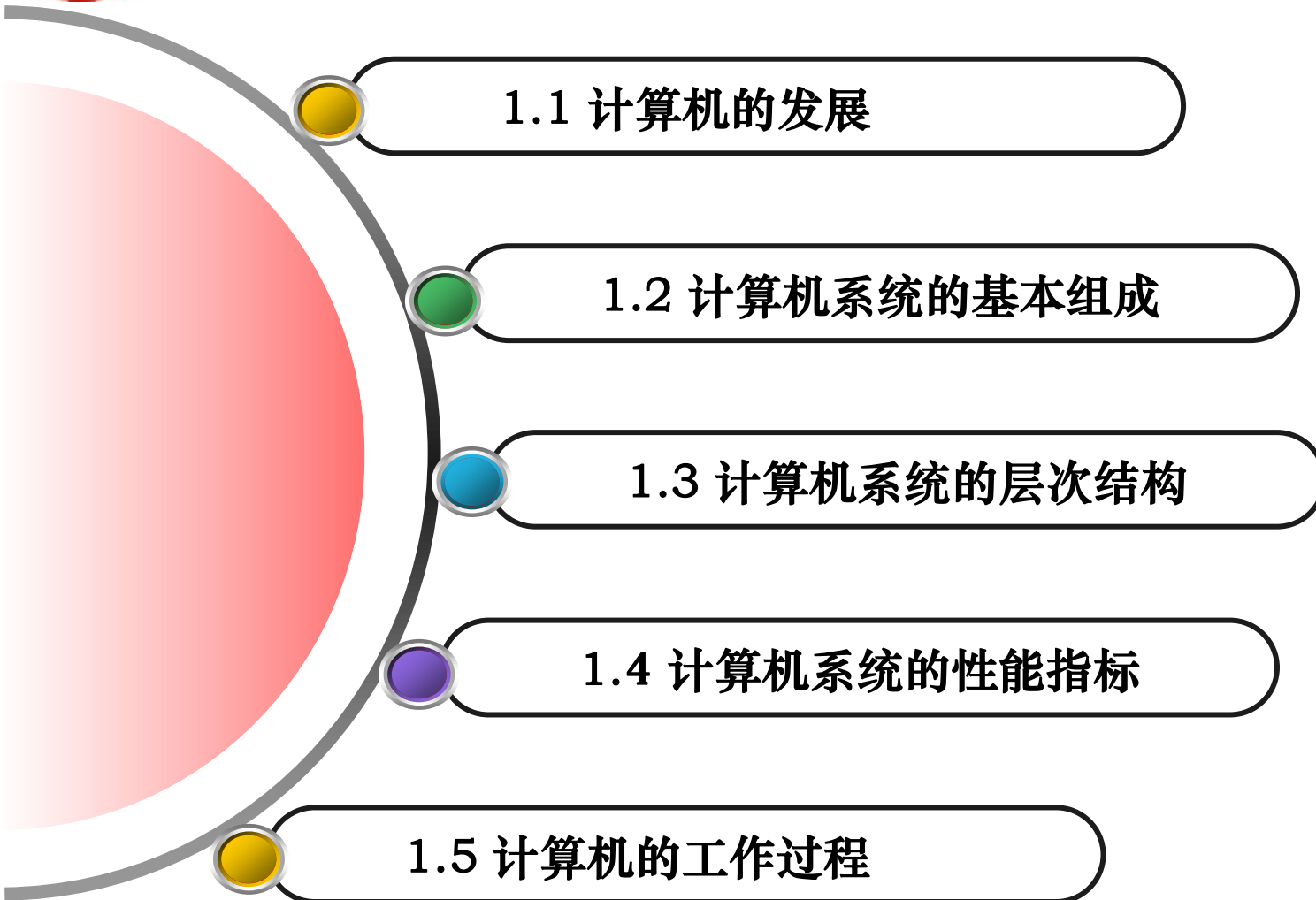


Andrew S. Tanenbaum



Randal E. Bryant  
David R.O'Hallaron

# 第一章结构



# 1.1 计算机的发展

1.1.1

计算机的发展历史

1.1.2

计算机的发展趋势

1.1.3

计算机的应用

# 计算机的发展历史

## 第一代电子管计算机（1946~1957）

- 逻辑元件：电子管，延迟线
- 运算速度：几次/秒

## 第二代晶体管计算机（1958~1964）

- 逻辑元件：晶体管，磁芯
- 运算速度：几十万次/秒

## 第三代集成电路计算机（1964~1972）

- 逻辑元件：中小规模IC(集成电路)，磁盘
- 运算速度：几百万次/秒

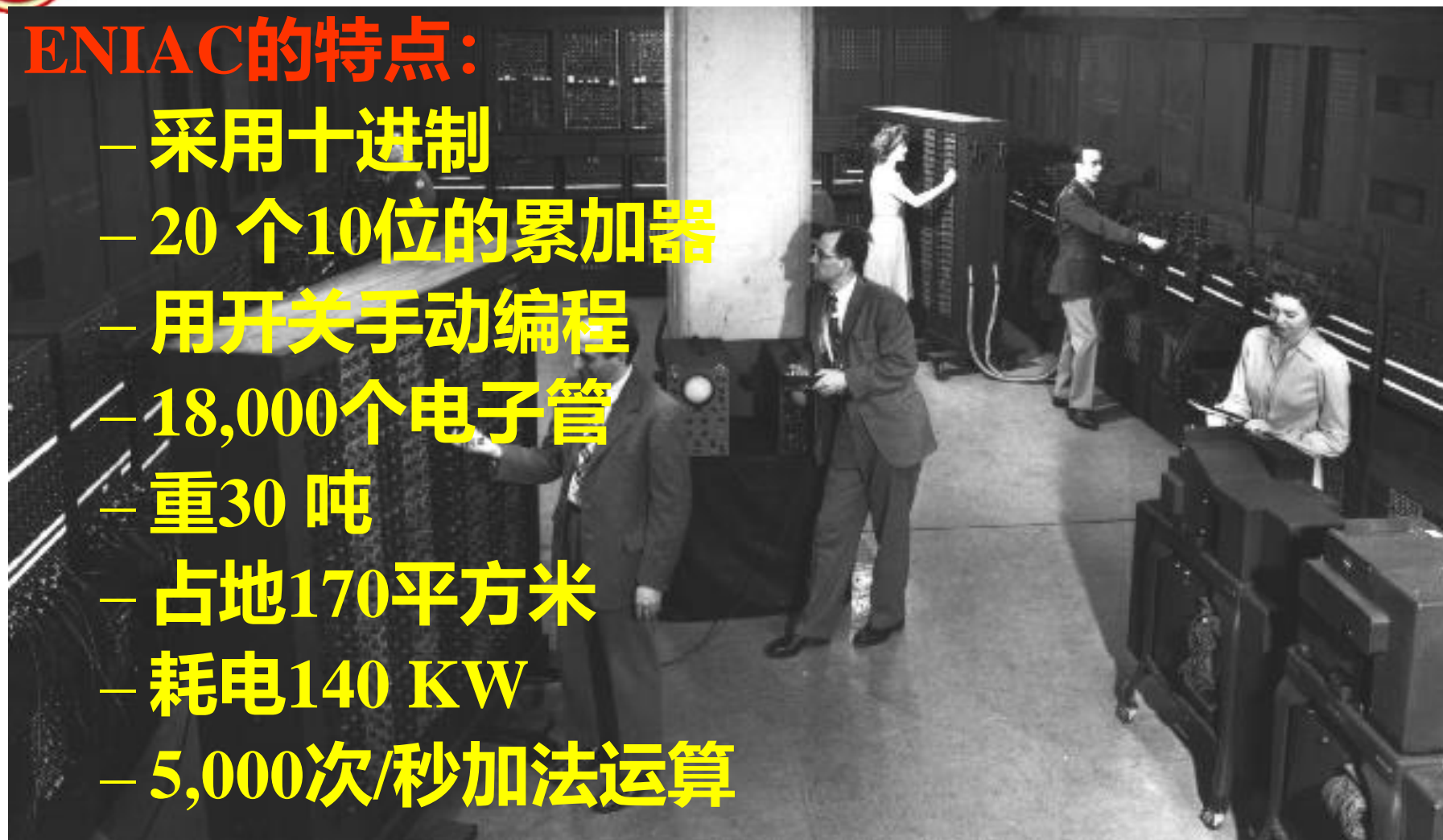
## 第四代大规模集成电路计算机（1972至今）

- 大、超大规模集成电路

# 第一台电子数字计算机ENIAC

## ENIAC的特点:

- 采用十进制
- 20 个10位的累加器
- 用开关手动编程
- 18,000个电子管
- 重30 吨
- 占地170平方米
- 耗电140 KW
- 5,000次/秒加法运算





# 第一台电子数字计算机ENIAC

机器	器件	加法运算速度	程序指令	用途
ENIAC	电子管	5000次/秒	硬件互联（连接线） 编制程序	专用（生产火炮 射表）

- ❑ ENIAC的缺点
  - 利用开关和连线设置程序指令，程序设计繁琐、低效
- ❑ 思考：
  - 将程序指令存储在机器中，由它控制机器的操作

# 存储程序原理



## □ 冯·诺依曼提出存储程序原理

### 和存储程序数字计算机

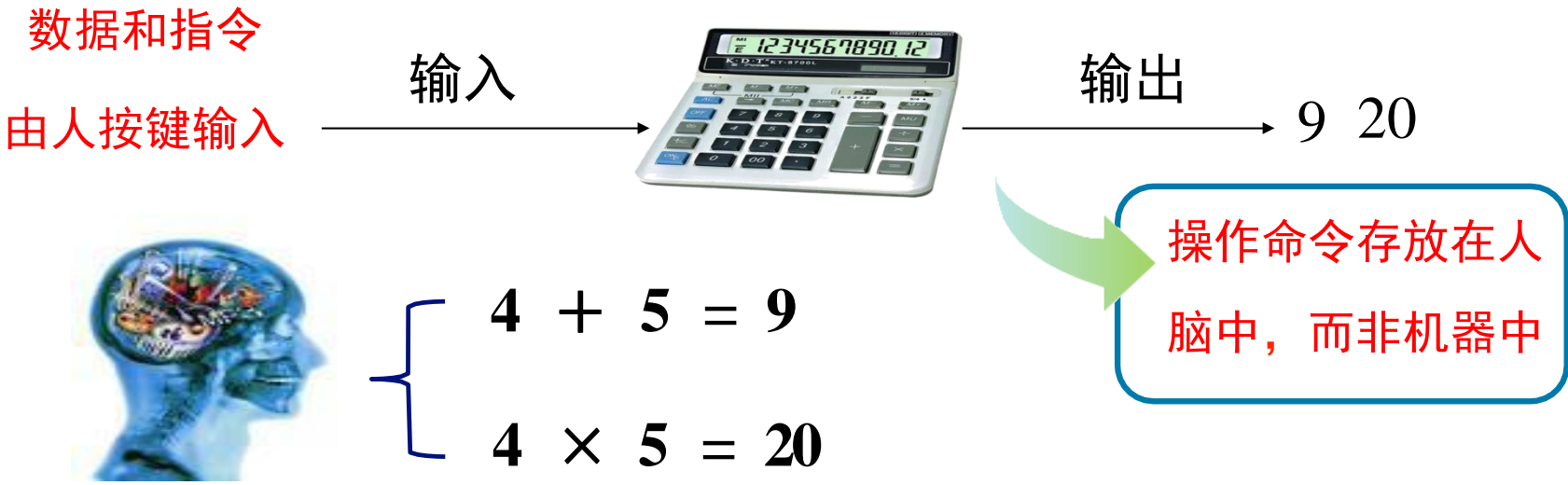
- 计算机（指硬件）应由运算器、存储器、控制器、输入设备和输出设备**五大基本部件组成**；
- 计算机内部采用**二进制**来表示指令和数据；
- 将编好的**程序**和原始**数据**事先**存入存储器**中，然后再启动计算机工作。

关键思想：存储程序



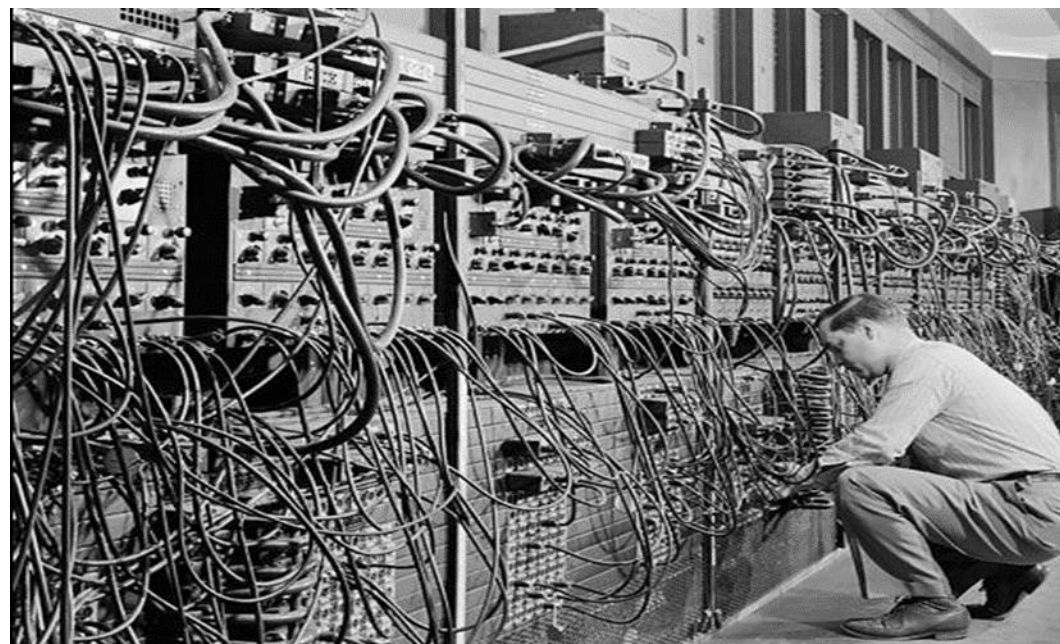
# 存储程序原理

□ 思考：如果不存储程序？



# 存储程序原理

□ 思考：如果不存储程序？



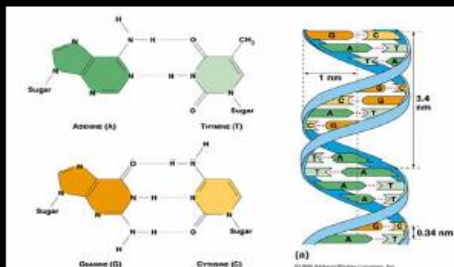


## 非冯·诺依曼结构模型

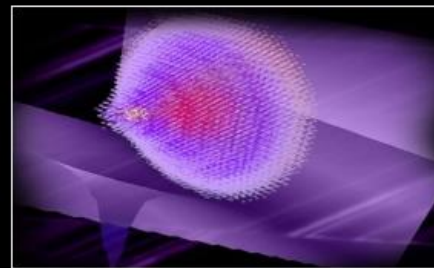
## 利用人脑模型思想作为计算范式



# 利用生物学和 DNA演化思想开 发算法



# 采用量子力学的 奇妙思想解决计 算问题



# 计算机的发展趋势-巨型化

- 超级计算机是一种高性能计算机，它集成数千乃至数百万个处理器核心，能够以极高的运算速度执行海量数据处理复杂的科学、工程计算任务。
- 应用于气候预测、基因组测序与分析、核聚变模拟、宇宙大爆炸理论模拟、量子物理、新材料研发等诸多前沿科技和基础科学研究领域。
- 1975年世界上第一台超级计算机“Cray-I”。
- TOP500已发展成为全世界最具权威的超级计算机排行榜，是衡量各国超级计算水平的最重要的参考依据。



# 计算机的发展趋势-巨型化

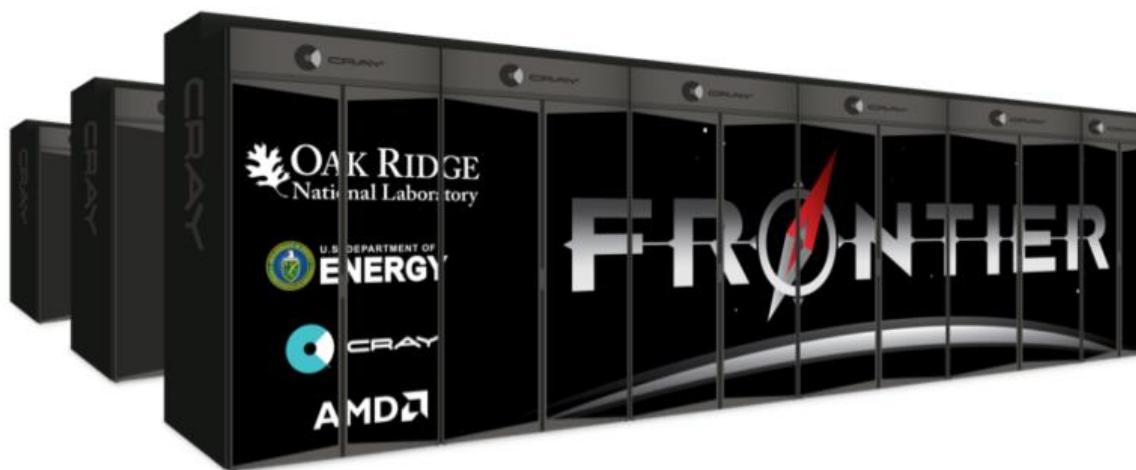
最权威的超级计算机排名的参考网址

<http://www.top500.org>



# 世界最快的超级计算机（截止到 2023.11）

美国橡树岭国家实验室 **Frontier**



- 以**1.194 EFlop/s**的实测性能保持领先地位，唯一的百亿亿次级超算；
- 搭载了基于最新的**HPE Cray EX235a**架构的**2GHz AMD EPYC 64C** 处理器，目前共有**8699904**个**CPU**和**GPU**核心；
- 具有高达**52.59 GFlops/W**的额定能效，并可借助于**HPE**的**Slingshot 11**网络进行数据传输。


# 我国最快的超级计算机（截止到 2023.11）

国家超级计算无锡中心的“神威·太湖之光”



- 完全采用中国设计和制造的处理器研制而成。
- 处理器：10,649,600个；峰值速度：93,015 TFlop/s。
- 包括40,960个节点，速度比“天河二号”快2倍，效率更是其3倍。

# 微型化

 计算机不再是单一的计算机器，而是一种信息处理机器，一种个人化的信息处理机器。

1975年，ATARI-8800微型机问世

1976年，苹果-I型微型机诞生

1979年，夏普公司宣布制成第一台手提式微型机

1982年，微型机开始普及，大量进入学校和家庭

# 通用微处理器



## ➤ 4位微处理器

- 1971年，Intel 4004，第一个微处理器

## ➤ 8位微处理器

- M6800、Z80和Intel 8080/8085
- Apple公司苹果机

## ➤ 16位微处理器

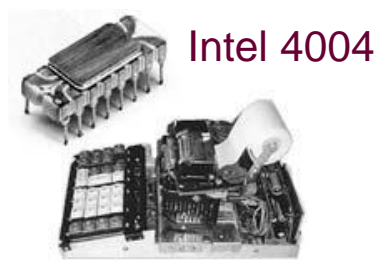
- Intel 8086/8088
- 16位个人计算机（PC：Personal Computer）

## ➤ 32位微处理器

- 80386，80486，Pentium～Pentium 4
- 32位PC机，APPLE公司的Macintosh机

## ➤ 64位微处理器

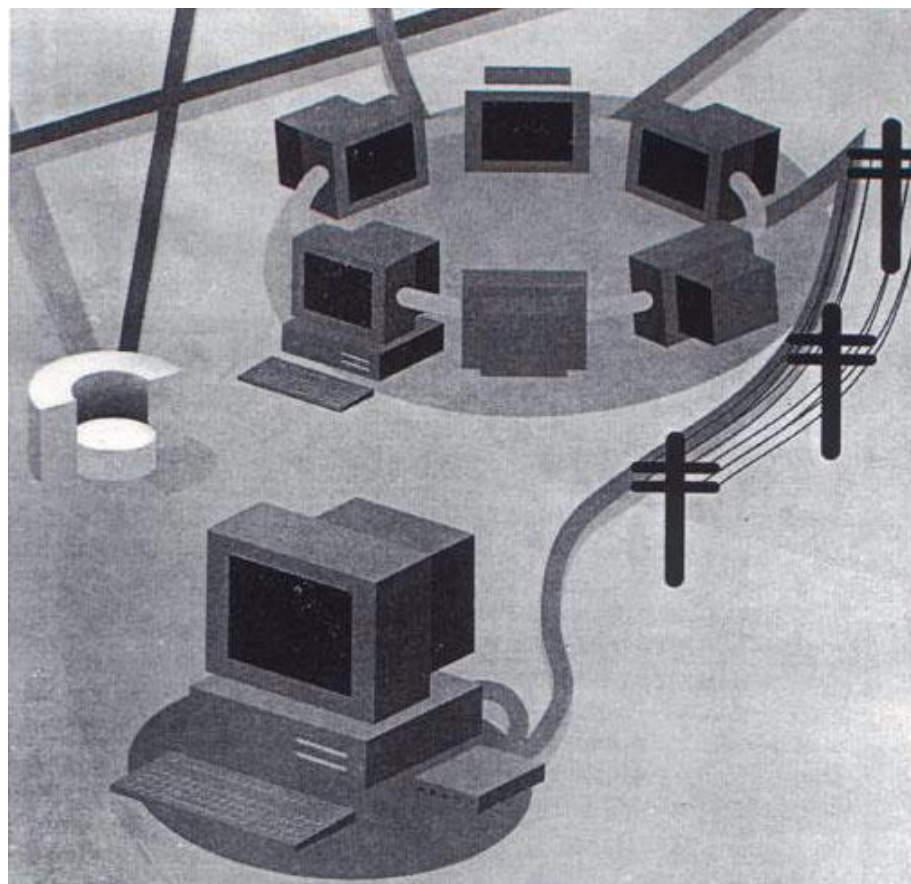
- IA-64结构：Itanium（安腾）
- x86结构：AMD的64位处理器、Intel 64处理器



采用4004的计算器



# 网络化



## 计算机网络：

计算机技术与通信技术紧密结合的产物。

## 计算机网络的发展动力：

使用远程资源，共享程序、数据和信息资源，网络用户的通讯和合作。

## 一种奇特的工具：

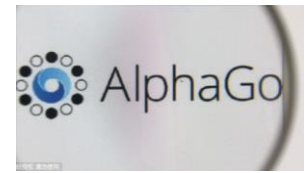
网络是基于全人类智慧的崭新的生产工具，越用越有价值。

# 智能化

“总有一天，人类会造出一些举止跟人一样的‘没有灵魂的机械’来”。

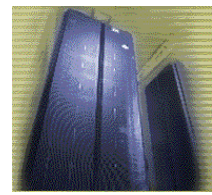
——笛卡尔（1637）

人类第一个“工业机器人”：  
一头在纺织机上挑纱的“驴”（1742年）



第一代机器人  
机械手（1962年出现）

第二代机器人  
具有“感觉”的机器人



Deepblue

第三代机器人  
装有启发式计算机的“智能机器人”



# 计算机的应用

## ■科学计算

密码破译，天气预报，地质勘探，卫星轨道计算

## ■过程控制

机器人以及各种自动化装备，温度调节，阀门控制

## ■辅助设计/分析/制造/教学

机械CAD，建筑CAD，CAE，CAM，CAI

## ■数据处理

数据库管理，企业信息管理，统计汇总、办公自动化

## ■智能模拟

人工智能、专家系统、自学习

# 小论文

## 课后作业：撰写小论文，题目任选其一

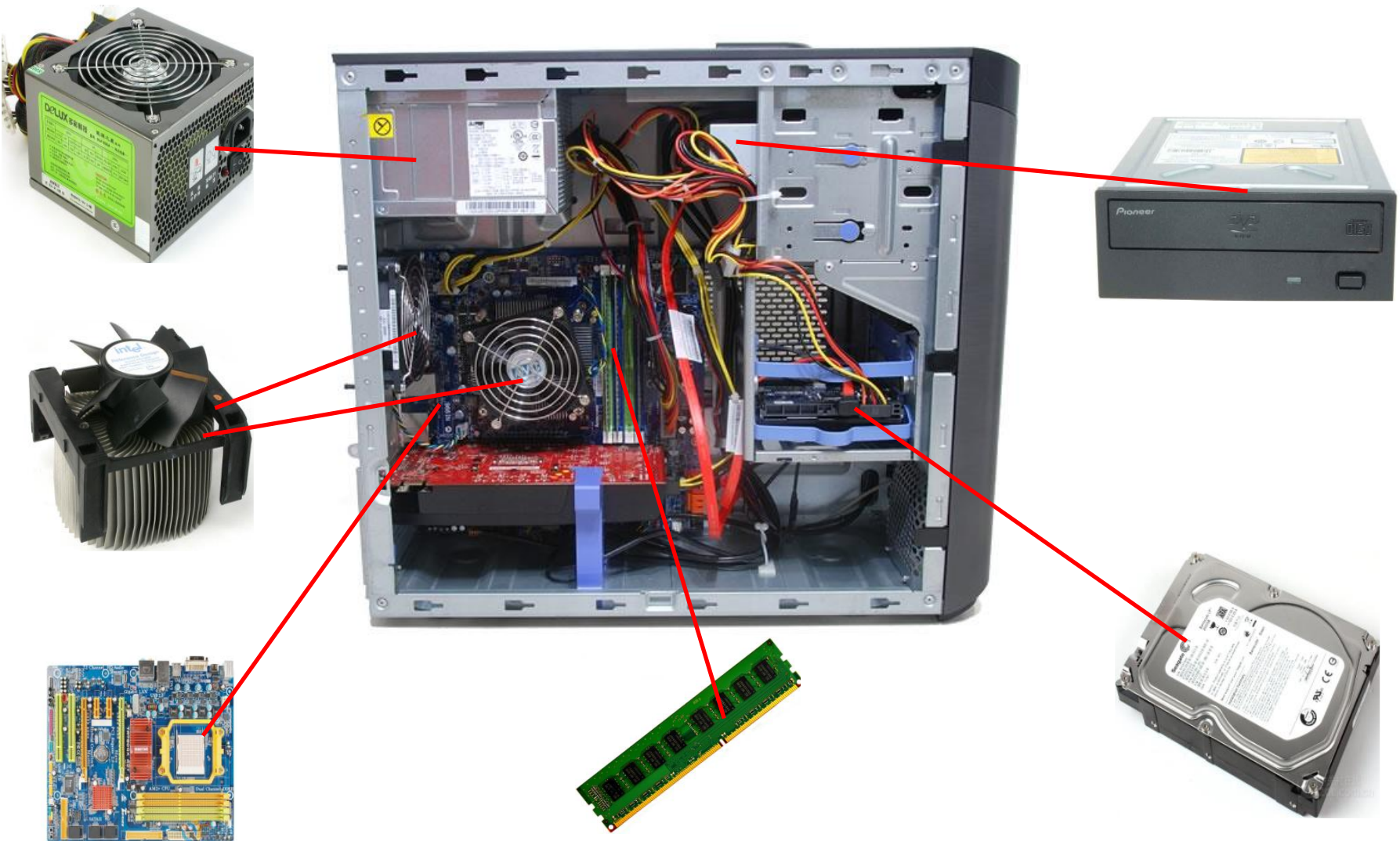
- 📖 题目一：计算机硬件技术的最新发展
- 📖 题目二：计算机技术发展与人工智能进步的相互影响及未来展望
- 📖 题目三：从共享存储到分布处理：非冯诺依曼计算机对传统计算模式的突破
- 📖 题目四：量子计算机及其对传统计算机系统结构的影响

# 1.2 计算机系统的基本组成



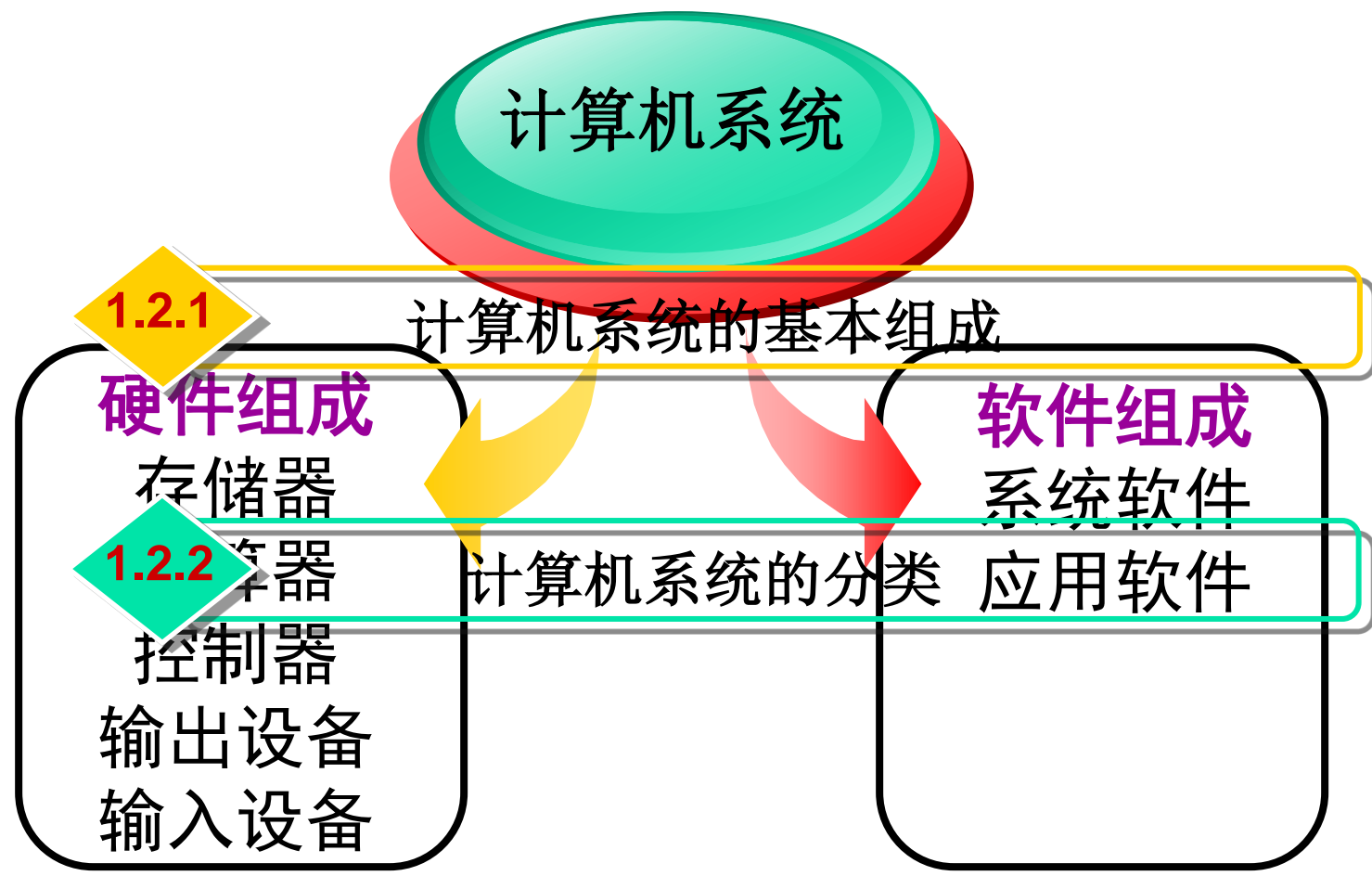
典型台式机

# 1.2 计算机系统的基本组成



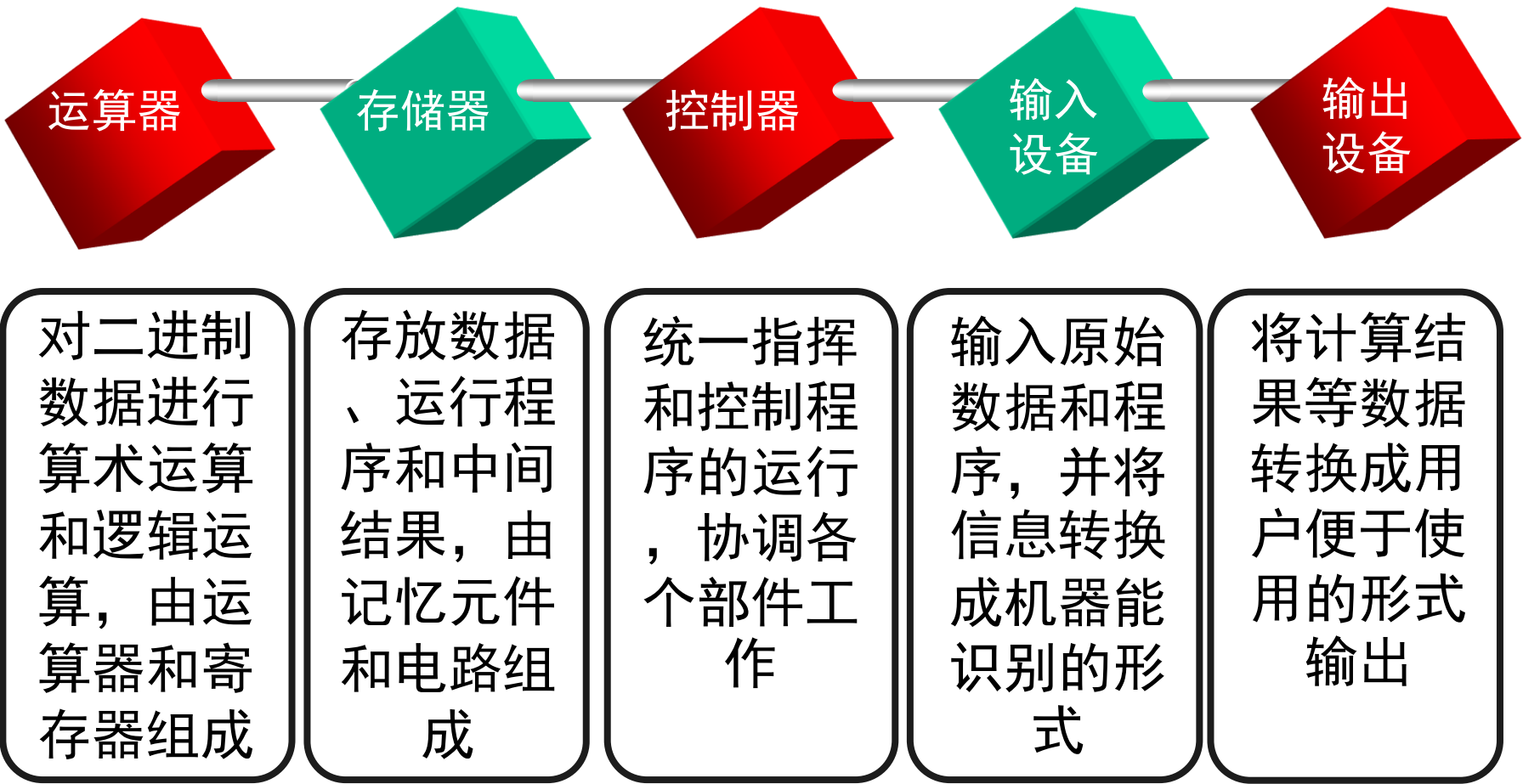


# 1.2 计算机系统的基本组成

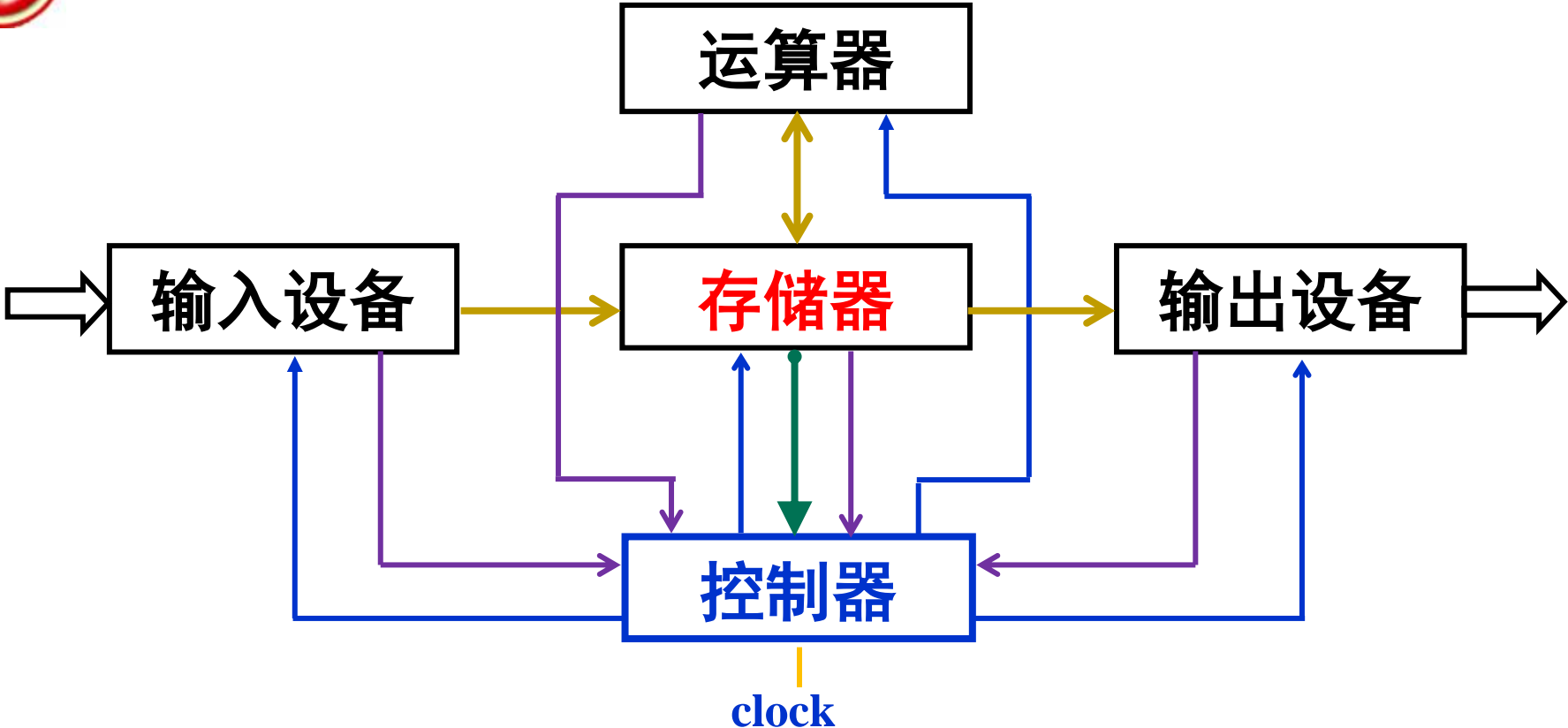


# 1.2.1 计算机系统的基本组成

## (一) 计算机的硬件组成



# 1.2.1 计算机系统的基本组成



运算器和控制器，已集成在CPU中！

- 绿色箭头线

→

指令信息
- 蓝色箭头线

→

控制信号
- 黄色箭头线

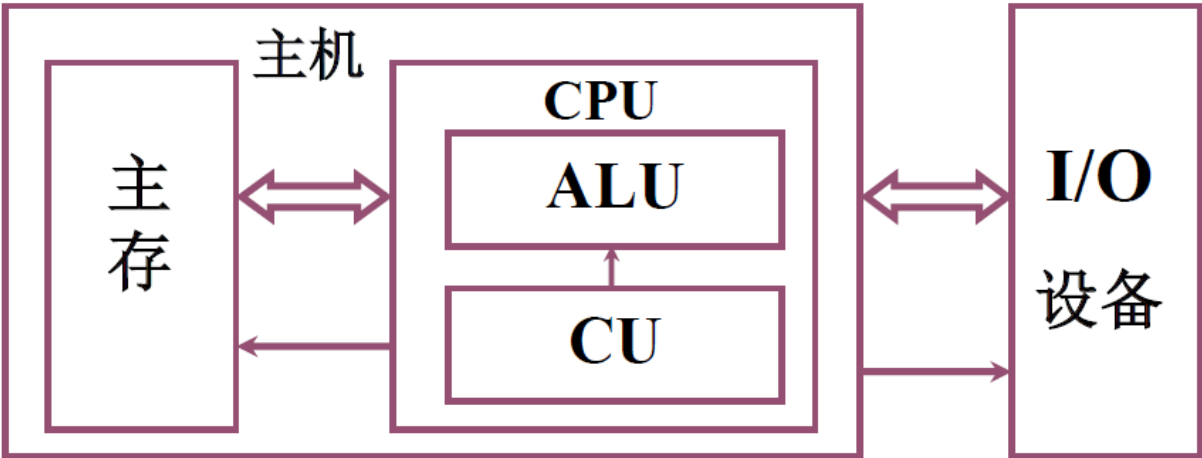
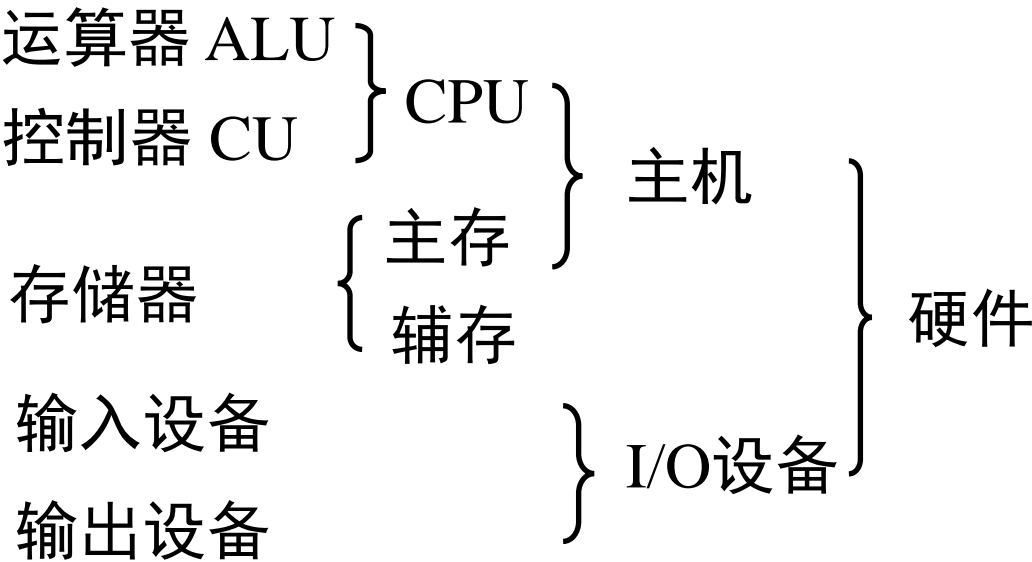
→

数据信息
- 紫色箭头线

→

状态信息

# 1.2.1 计算机系统的基本组成



## 1.2.1 计算机系统的基本组成

### (三) 计算机系统的软件组成

1

#### 系统软件

系统程序，管理系统、监视服务、合理调度系统资源。包括标准程序库、语言处理程序、操作系统、服务性程序、数据库管理系统、网络软件等。

2

#### 应用软件

应用软件又称为应用程序，它是用户根据任务需要所编制的各种程序，是用户在各自己的行业中开发和使用的各种程序。

## 1.2.2 计算机的分类

### ■ 按应用特点分类

#### 专用 计算机

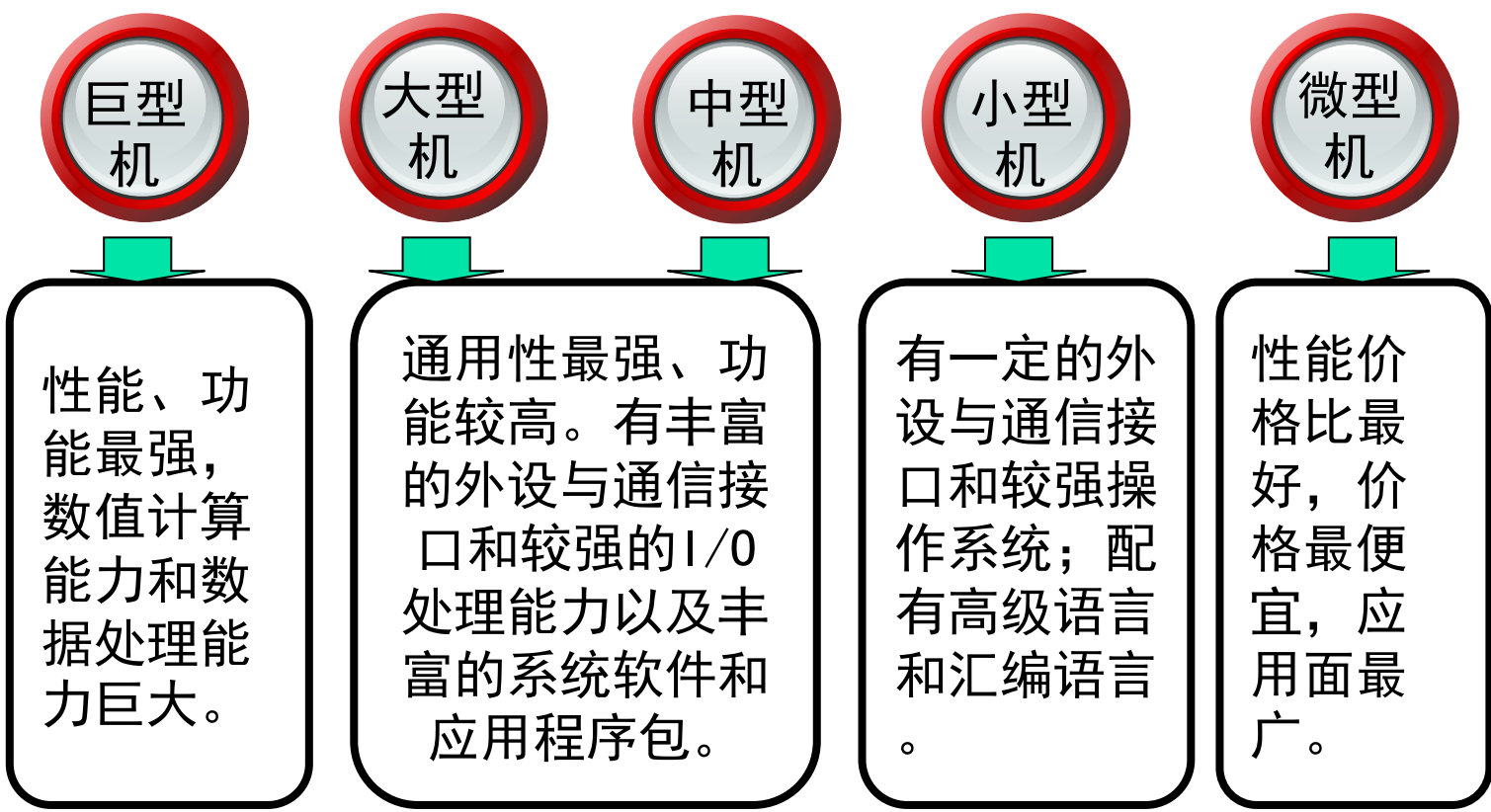
针对某一特定应用领域或面向某种算法而研制的计算机。对指定的领域有较高的性价比；其它领域则低效。

#### 通用 计算机

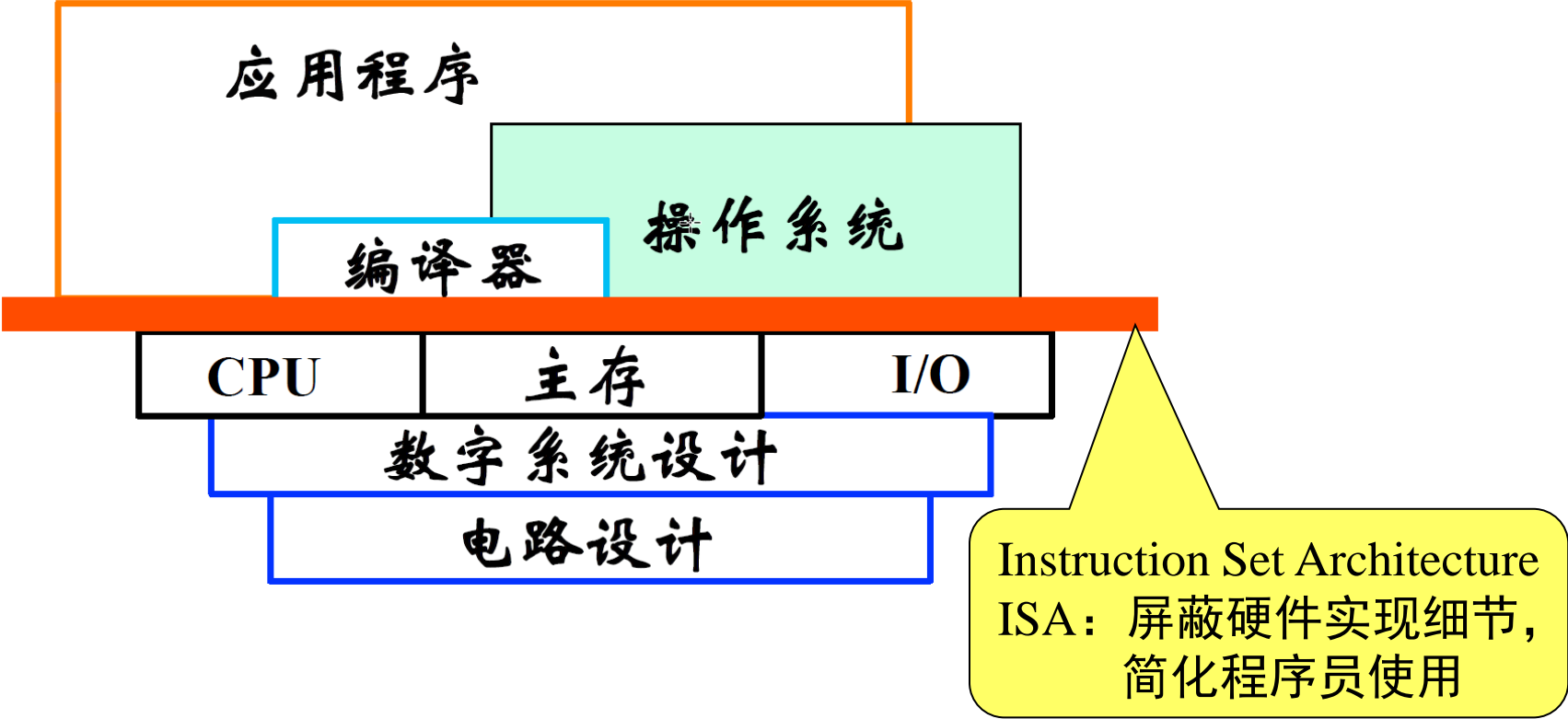
针对多种应用领域或面向多种算法而研制的计算机。有较复杂的软件及系统结构和较强的系统功能。

# 1.2.2 计算机系统的分类

## ■ 按性能特点分类

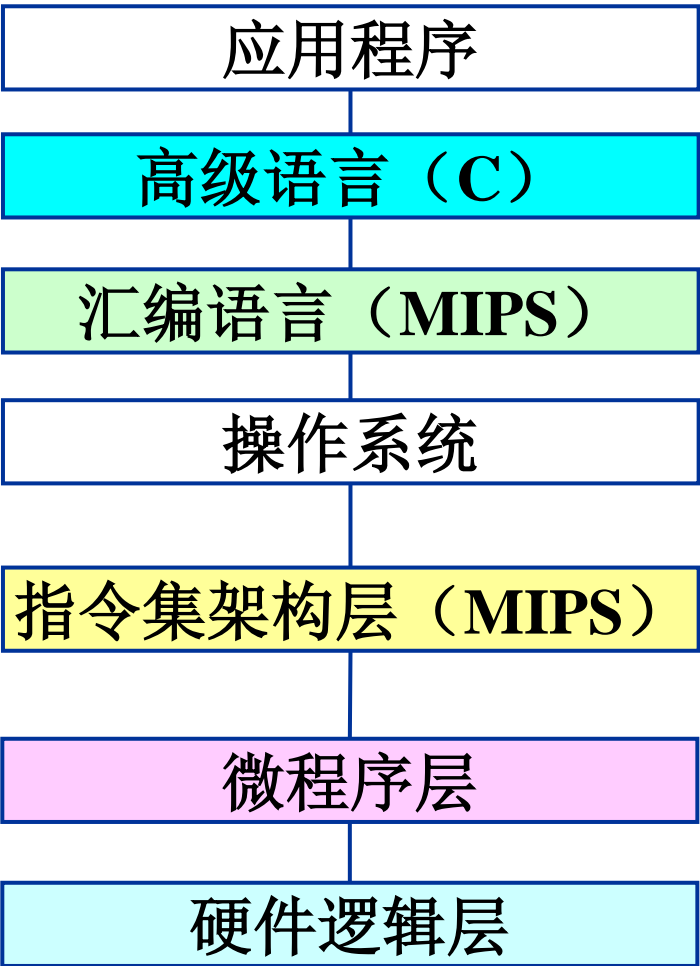


# 1.3 计算机系统的层次结构



- ❑ 计算机系统层次结构中，指令系统是软/硬件的交界面
- ❑ 不同用户工作在不同层次，看到的计算机是不一样的

# 1.3 计算机系统的层次结构



```
temp = v[k];  
v[k] = v[k+1];  
v[k+1] = temp;
```

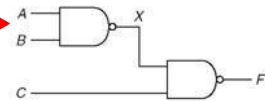
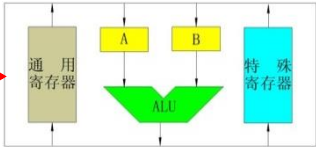
```
lw $t0, 0($2)  
lw $t1, 4($2)  
sw $t1, 0($)  
sw $t0, 4($2)
```

```
0000 1001 1100 0110 1010 1111 0101 1000  
1010 1111 0101 1000 0000 1001 1100 0110  
1100 0110 1010 1111 0101 1000 0000 1001  
0101 1000 0000 1001 1100 0110 1010 1111
```

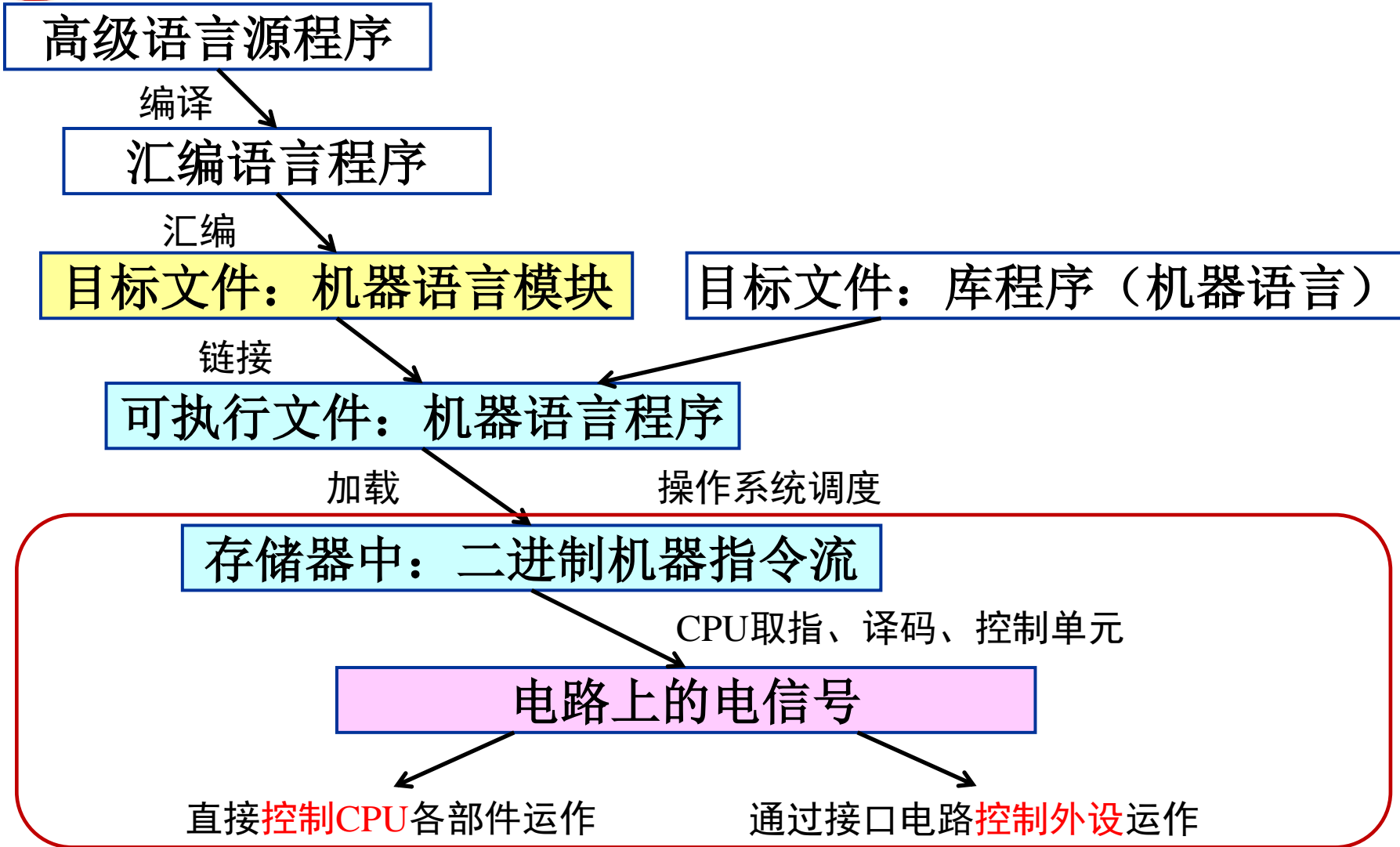
```
RWrite=1,Branch=Jump=0  
RegDst=0,ALUop=00110,  
Jump0.....
```

使用布尔代数和真值表

- 不同用户处在不同层次
- 不同层次具有不同属性
- 不同层次使用不同工具
- 不同层次代码效率不同



# 1.3 计算机系统的层次结构



## 1.4 计算机系统的性能指标

### □ 衡量计算机性能的基本指标

Time to do the task

□ 响应时间(Response Time): 完成单个任务所需的总时间

□ 执行时间(Execution Time)、等待时间(Latency)

□ 吞吐率(Throughput ): 单位时间内所完成的任务总量

□ 带宽(Bandwidth)

Tasks per sec

□ 指令执行速度(MIPS、MFLOPS)

### □ 不同应用场合用户关心的性能不同

□ 吞吐率高的场合—多媒体应用(音/视频播放要流畅)

□ 响应时间短的场合—事务处理系统(响应速度要快)

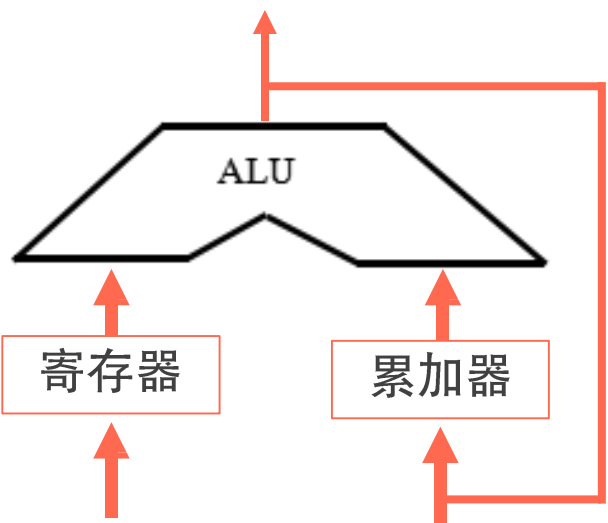
□ 吞吐率高且响应时间短的场合—ATM、文件服务器、Web服务器等

# 1.4 计算机系统的性能指标

## 1. 非时间指标

### (1) 机器字长

机器一次能处理的二进制的位数



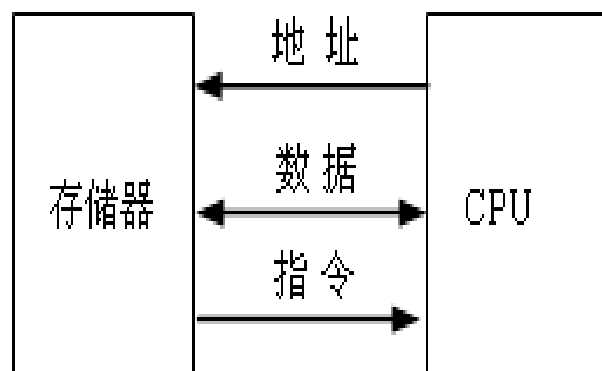
- 由加法器、寄存器的位数决定
- 一般与内部寄存器的位数相等 (字长)
- 字长越长，表示数据的范围就越大，精确度越高
- 目前常见的有32位和64位字长

## 1.4 计算机系统的性能指标

### 1. 非时间指标

#### (2) 总线宽度

数据总线一次能并行传送的最大信息的位数



- 一般指运算器与存储器之间的数据总线位数。
- 有些计算机内部与外部数据总线宽度不一致

## 1.4 计算机系统的性能指标

### 1. 非时间指标

#### (3) 主存容量与存储带宽

##### □ 主存容量

- 主存存放二进制信息的总位数
- 存储单元数  $\times$  存储字长

##### □ 存储带宽

- 单位时间内与主存交换的二进制信息量
- 常用单位B/s（字节/秒）。
- 影响存储带宽的指标包括数据位宽和数据传输速率

## 1.4 计算机系统的性能指标

### 1. 非时间指标

#### (3) 主存容量与存储带宽

##### □ 主存容量

- 主存存放二进制信息的总位数
- 存储单元数  $\times$  存储字长

##### □ 存储带宽

- 单位时间内与主存交换的二进制信息量
- 常用单位B/s（字节/秒）。
- 影响存储带宽的指标包括数据位宽和数据传输速率

## 1.4 计算机系统的性能指标

### 2. 时间指标

#### (1) 主频 $f$ /时钟周期 $T$ , 外频、倍频

##### □ 主频 $f$

- CPU内核工作的时钟频率

- 与CPU实际的运算能力之间不是唯一的、直接关系

##### □ 时钟周期 $T$

- 节拍周期, 是计算机中最基本的、最小的时间单位

- 在一个时钟周期内, CPU仅完成一个最基本的动作

##### □ $f$ 与 $T$ 的关系

- 互为倒数,  $f$  越高,  $T$  就越小

## 1.4 计算机系统的性能指标



**外频：** CPU（内存）与主板之间同步的时钟频率  
（系统总线的工作频率）



**倍频：** CPU主频与外频之间的倍数

主频 = 外频 × 倍频

例如：Pentium 4 2.4G CPU主频

$2400\text{M} = 133\text{M} (\text{外频}) \times 18 (\text{倍频})$

## 1.4 计算机系统的性能指标

### 2. 时间指标

#### (2) CPI (Clock cycles Per Instruction)

- ❑ 执行一条指令(平均)需要的时钟周期数, 即T周期的个数
- ❑ 单条指令CPI、一段程序中所有指令的CPI、指令系统CPI等

$$\begin{aligned} \text{CPI} &= \text{程序中所有指令的时钟周期数之和} / \text{程序指令总数} \\ &= \sum (\text{程序中各类指令的CPI} \times \text{程序中该类指令的比例}) \end{aligned}$$

# 1.4 计算机系统的性能指标

## 2. 时间指标

### CPI 应用举例

**【例1】** 某计算机指令系统中各类指令所占比例及CPI如下表所示，求程序的CPI即T周期的个数

指令类型	CPI	指令比例
算术和逻辑	1	60%
Load/Store	2	18%
转移	4	12%
Cache缺失访存	8	10%

$$CPI = 1*60\% + 2*18\% + 4*12\% + 8*10\% = 2.24$$

## 1.4 计算机系统的性能指标

### 2. 时间指标

#### (2) IPC (Instructions Per Clock cycle )

每个时钟周期执行的指令条数(并行)

#### (3) MIPS (Million Instructions Per Second)

每秒钟CPU能执行的指令总条数 (单位：百万条/秒)

$$\begin{aligned} \text{MIPS} &= \frac{\text{指令条数}}{\text{执行时间} \times 10^6} = \frac{\text{指令条数}}{(\text{所有指令CPU时钟周期数之和} / f) \times 10^6} \\ &= \frac{f}{\text{CPI} \times 10^6} \end{aligned}$$

#### (4) FLOPS, 每秒执行浮点运算的次数

如, 天河2号, 其实测速度: 33.86PFLOPS

# 1.4.2 计算机系统评价标准

【例2】某计算机主频是1.2GHz，其指令分为4类，它们在基准程序中所占比例及CPI如下表所示，求该机器的MIPS

。

指令类型	所占比例	CPI
A	50%	2
B	20%	3
C	10%	4
D	20%	5

$$CPI=50\% \times 2+20\% \times 3+10\% \times 4+20\% \times 5=3$$

$$MIPS=f/(CPI \times 10^6)=1200/3=400$$

## 1.4 计算机系统的性能指标

### 2. 时间指标

#### (5) CPU时间

□ 执行一段程序所需的时间

(CPU时间+ I/O时间 + 存储访问时间+ 各类排队时延等)。

$$\begin{aligned}\text{CPU时间} &= \text{程序中所有指令的时钟周期数之和} \times T \\ &= \text{程序中所有指令的时钟周期数之和} / f\end{aligned}$$

## 1.4 计算机系统的性能指标

### 2. 时间指标

#### CPU时间的计算方法

#### □ 考虑CPI后的CPU时间

指令数 × CPI × 时钟周期

$$\text{CPU时间} = \text{总指令数} \times \sum_{i=1} (CPI_i \times \frac{IC_i}{\text{总指令数}}) \times \text{时钟周期时间}$$

$$= \text{总指令数} \times \text{CPI} \times T$$

#### □ 考虑MIPS后的CPU时间

$$\text{MIPS} = \frac{\text{指令数}}{\text{执行指令的时间} \times 10^6}$$

$$\text{程序的执行时间 } t = \frac{\text{指令数量}}{\text{MIPS} \times 10^6}$$

# 1.4 计算机系统的性能指标

【例3】某计算机主频为1GHZ，在其上运行的目标代码包含 $2 \times 10^5$ 条指令，分4类，各类指令所占比例和各自CPI如下表所示，求该段程序的CPU时间。

指令类型	CPI	指令混合比例
算术和逻辑	1	60%
Load/Store	2	18%
转移	4	12%
Cache缺失访存	8	10%

方法1：利用CPI

$$\begin{aligned} \text{CPU时间} &= 2 \times 10^5 \times \text{CPI} / f \\ &= (2 \times 10^5 \times 2.24 / 10^9) = 4.48 \times 10^{-4} \text{ (秒)} \end{aligned}$$

方法2：利用MIPS

$$\text{CPU时间} = \frac{\text{指令数量}}{\text{MIPS} \times 10^6} = \frac{2 \times 10^5}{(10^3 / 2.24) \times 10^6} = 4.48 \times 10^{-4} \text{ (秒)}$$

## 1.4 计算机系统的性能指标

**【例4】** 假设计算机M的指令集中包含A、B、C三类指令，其中CPI分别是1、2、4。某个程序P在M上被编译成两个不同的目标代码序列P1和P2，P1所含A、B、C三类指令的条数分别为8、2、2，P2所含A、B、C三类指令的条数分别为2、5、3。请问：哪个代码序列指令条数少？哪个执行速度快？它们的CPI分别是多少？

(1) P1和P2的指令条数为12和10，所以P2的指令条数少；

(2) 执行P1所需时间： $(8 \times 1 + 2 \times 2 + 2 \times 4) \times T = 20T$

执行P2所需时间： $(2 \times 1 + 5 \times 2 + 3 \times 4) \times T = 24T$

同一机器，f相同，T相同，所以P1比P2快

(3)  $\text{CPI} = \text{总时钟周期数} / \text{总指令条数}$

$\text{CPI}_1 = 20 / 12 = 1.67$      $\text{CPI}_2 = 24 / 10 = 2.4$

$\text{MIPS} = \text{指令条数} / \text{执行时间}$   
 $= f / \text{CPI}$

◆ 评价计算机性能，要综合考虑多个因素。



# 1.4 计算机系统的性能指标

CPU时间 = 指令数 × CPI × 时钟周期

【例5】一个程序在一台时钟频率4GHz的计算机A上运行10秒，现在我们要设计一台时钟频率更高的计算机B，将该程序在计算机B上的执行时间降低为6秒。提高时钟频率将影响CPU其他方面的设计，使得在计算机B上运行该程序的时钟周期数增长为1.2倍。请分析计算机B的时钟频率？

	机器A	机器B
CPU时钟频率	4GHz	? ?
程序运行的时钟周期数	I	I × 1.2 ↑
执行时间	10秒	6秒 ↓

CPU时间<sub>A</sub> = CPU时钟周期数<sub>A</sub> / 时钟频率<sub>A</sub>  
→ CPU时钟周期数<sub>A</sub> = 10 sec × 4 GHz

时钟频率<sub>B</sub> = CPU时钟周期数<sub>B</sub> / CPU时间<sub>B</sub>  
= 1.2 × 10sec × 4GHz / 6sec = 8GHz

机器B的时钟频率是A的两倍，但机器B的速度并不是A的两倍！



## 1.4 计算机系统的性能指标

**CPU时间 = 指令数 × CPI × 时钟周期**

【例6】假设有相同指令级的两种不同实现方式：计算机A的时钟周期为250ps，对某程序的CPI为2.0；计算机B的时钟周期为500ps，对同样程序的CPI为1.2.请问：对于该程序，请问哪台计算机执行的速度更快？快多少？

**相同指令集对于同一个程序，其指令序列一样，指令数相同**

$$\begin{aligned}\text{CPU时间}_A &= \text{指令数} \times \text{CPI}_A \times \text{时钟周期}_A \\ &= I \times 2.0 \times 250\text{ps} \\ &= I \times 500\text{ps}\end{aligned}$$

$$\begin{aligned}\text{CPU时间}_B &= \text{指令数} \times \text{CPI}_B \times \text{时钟周期}_B \\ &= I \times 1.2 \times 500\text{ps} \\ &= I \times 600\text{ps}\end{aligned}$$

$$\begin{aligned}\text{CPU性能}_A / \text{性能}_B &= \text{执行时间}_B / \text{执行时间}_A \\ &= (I \times 600\text{ps}) / (I \times 500\text{ps}) \\ &= 1.2\end{aligned}$$

# 1.4 计算机系统的性能指标

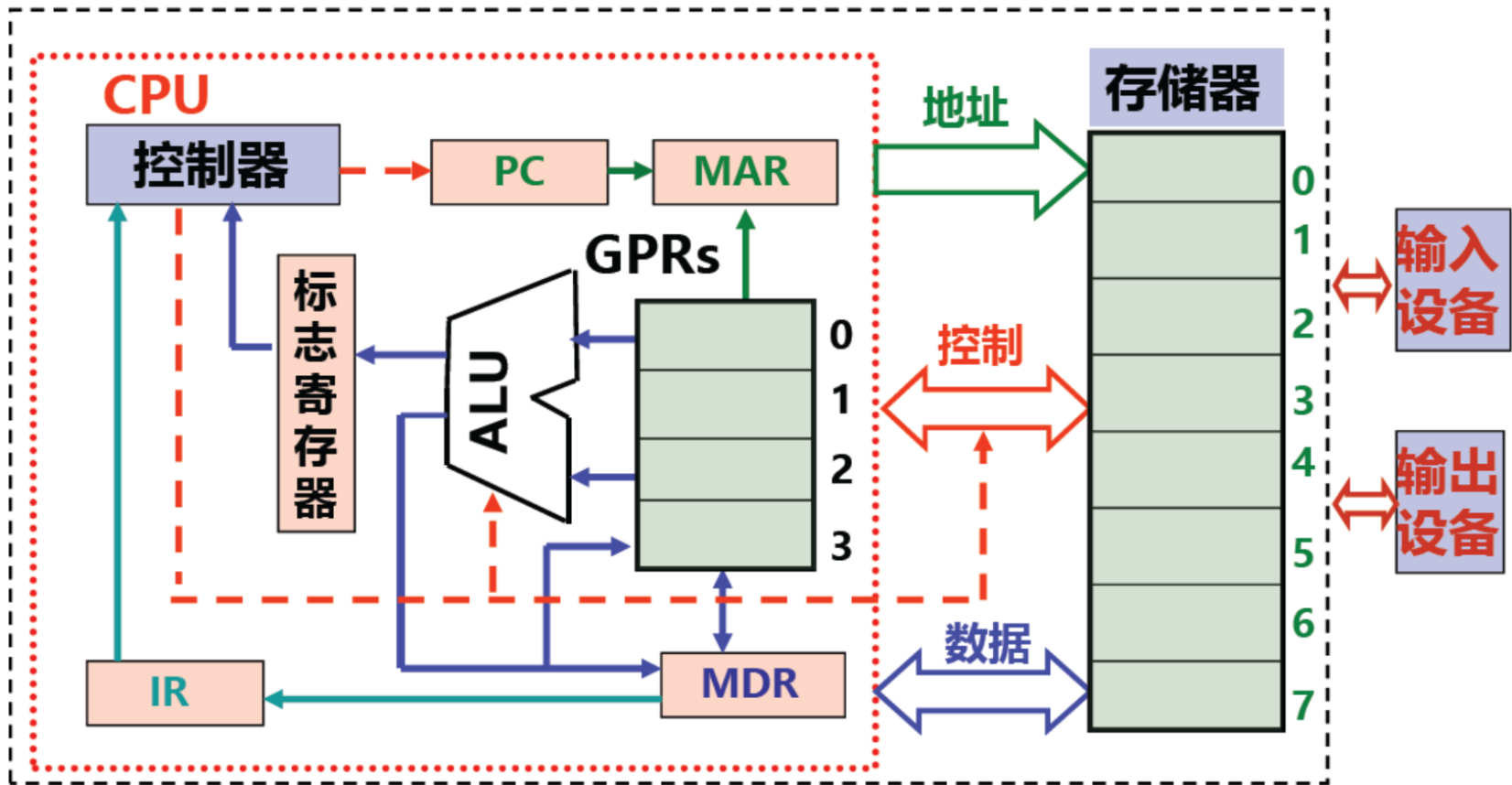
## 理解程序性能

指标	影响因素	如何影响
算法	指令数、CPI	算法决定源程序的指令条数，因此决定了处理器执行的指令条数。算法由于对慢速或快速指令的不同倾向性而同样影响了CPI。例：若算法更多地使用浮点运算，将获得更高的CPI
编程语言	指令数、CPI	由于编程语言中的语句被翻译成CPU指令，而后者决定了指令条数，因此编程语言势必会影响指令条数。由于编程语言的自身特点，它可能同样影响CPI。例：强支持数据抽象的语言(如Java)要求间接调用，而这往往会用到具有高CPI的指令
编译程序	指令数、CPI	编译器的效率对指令数及每条指令的平均周期数均有影响，这是因为编译器决定了从源语言语句到机器指令的翻译。编译器的作用可以非常复杂，同时通过复杂的方式影响CPI
指令集体系结构	指令数、时钟频率、CPI	指令集的结构对CPU性能的这三个方面均有影响，因为它影响了完成某功能需要的指令数、每条指令周期数及CPU时钟频率



# 1.5 计算机的工作过程

你还记得冯.诺依曼计算机结构的特点吗？



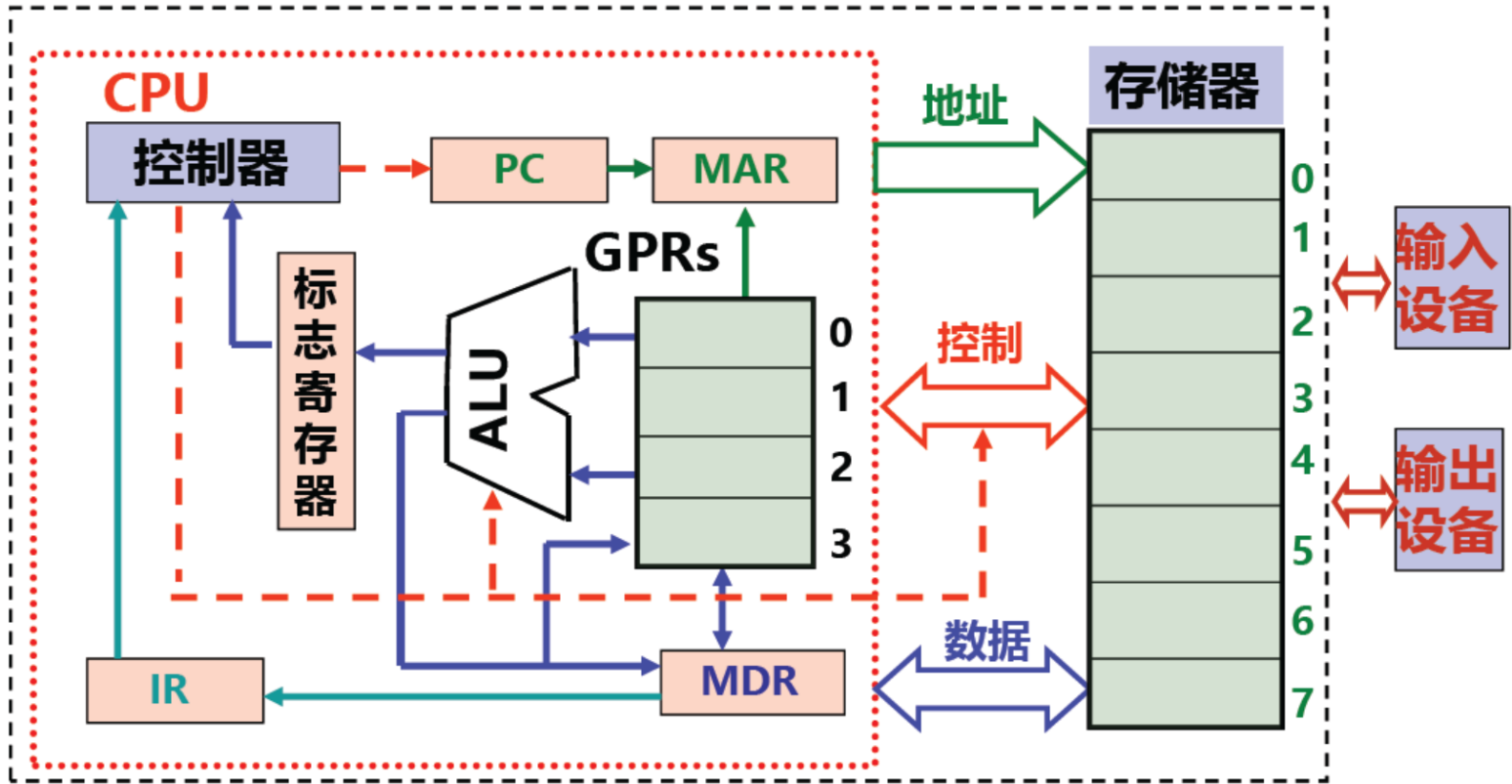
你能想到计算机相当于现实生活中的什么呢？ 工厂、饭店？

计算机是如何工作的呢？



# 1.5 计算机的工作过程

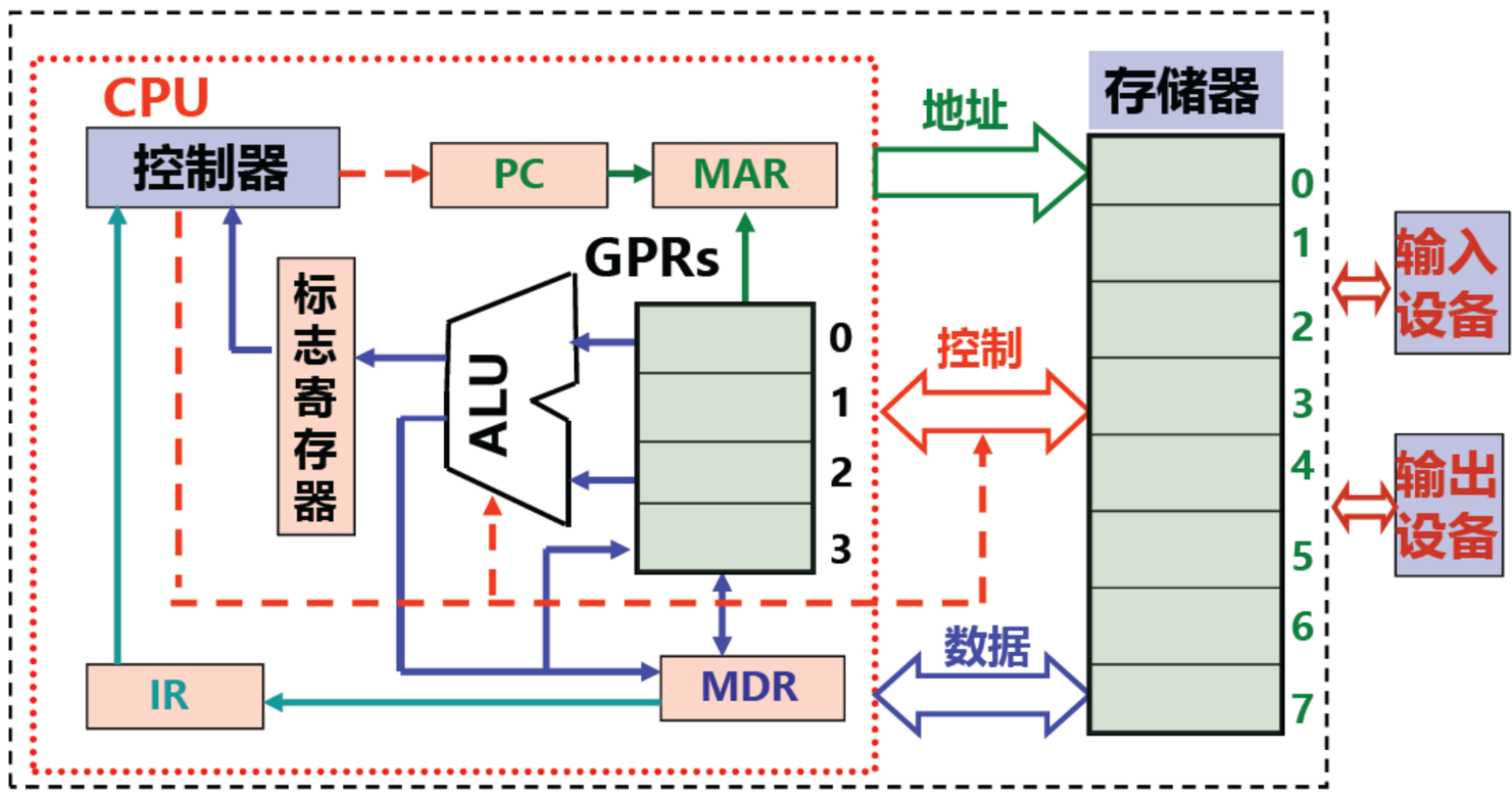
**CPU**：中央处理器；**PC**：程序计数器；**MAR**：存储器地址寄存器  
**ALU**：算术逻辑部件；**IR**：指令寄存器；**MDR**：存储器数据寄存器  
**GPRs**：通用寄存器组（由若干通用寄存器组成，早期就是累加器）



# 1.5 计算机的工作过程

想象一下妈妈是如何给你做一桌你喜欢（指定）的菜的？

厨房-CPU，妈妈-控制器，盘-GPRs，锅灶等-ALU，架子-存储器



## 1.5 计算机的工作过程



### • 做菜前

类似“存储程序”工作方式

- 原材料（**数据**）和菜谱（**指令**）都**按序**放在厨房外的架子（**存储器**）上，每个架子有编号（**存储单元地址**）。
- 菜谱上信息：原料位置、做法、做好的菜放在哪里等
  - 例如，把10、11号架上的原料一起炒，并装入3号盘
- 你告诉妈妈从第5个架上（**起始PC=5**）指定菜谱开始做

### • 开始做菜

- 第一步：从5号架上取菜谱（**根据PC取指令**）
- 第二步：看菜谱（**指令译码**）
- 第三步：从架上或盘中取原材料（**取操作数**）
- 第四步：洗、切、炒等具体操作（**指令执行**）
- 第五步：装盘或直接送桌（**写回结果**）
- 第六步：算出下一菜谱所在架子号 $6=5+1$ （**修改PC的值**）

### • 继续做下一道菜（**执行下一条指令**）




## 1.5 计算机的工作过程

### 程序由指令组成（菜单由菜谱组成）

- 程序在执行前
- 数据和指令事先存放在存储器中，每条指令和每个数据都有地址，指令按序存放，指令由OP、ADDR字段组成，程序起始地址置PC
- （原材料和菜谱都放在厨房外的架子上，每个架子有编号。妈妈从第5个架上指定菜谱开始做）
- 开始执行程序
  - 第一步：根据PC取指令（从5号架上取菜谱）
  - 第二步：指令译码（看菜谱）
  - 第三步：取操作数（从架上或盘中取原材料）
  - 第四步：指令执行（洗、切、炒等具体操作）
  - 第五步：写回结果（装盘或直接送桌）
  - 第六步：修改PC的值（算出下一菜谱所在架子号 $6=5+1$ ）
- 执行下一条指令（继续做下一道菜）

## 1.5 计算机的工作过程

- 
- 📖 程序启动前，指令和数据都存放在存储器中，形式上没有差别，都是0/1序列
  - 📖 采用“**存储程序**”工作方式：
    - ✧ 程序由指令组成，程序被启动后，计算机能自动取出一条一条指令执行，在执行过程中无需人的干预。
  - 📖 指令执行过程中，指令和数据被从存储器取到CPU，存放在CPU内的寄存器中，指令在IR中，数据在GPR中。
  - 📖 指令中需给出的信息
    - ✧ 操作性质（操作码）
    - ✧ 源操作数1 或/和 源操作数2（立即数、寄存器编号、存储地址）
    - ✧ 目的操作数地址（寄存器编号、存储地址）

计算机实现的所有任务都是通过执行一条一条指令完成的！



# 自学：计算机的工作步骤

## 1. 上机前的准备

- 建立数学模型
- 确定计算方法

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \frac{x^9}{9!} - \dots$$

$$\sqrt{x} = \frac{1}{2} \left( y_n + \frac{x}{y_n} \right) \quad (n = 0, 1, 2, \dots)$$

- 编制解题程序

程序 —— 运算的 全部步骤

指令 —— 每 一个步骤



编程举例

计算

$ax^2 + bx + c = (ax + b)x + c$

取 $x$  至运算器中

乘以 $x$  在运算器中

乘以 $a$  在运算器中

存 $ax^2$  在存储器中

取 $b$  至运算器中

乘以 $x$  在运算器中

加 $ax^2$  在运算器中

加 $c$  在运算器中

取 $x$  至运算器中

乘以 $a$  在运算器中

加 $b$  在运算器中

乘以 $x$  在运算器中

加 $c$  在运算器中

# 指令格式举例

操作码	地址码
-----	-----

取数	$\alpha$	$[\alpha] \rightarrow \text{ACC}$
000001	0000001000	
存数	$\beta$	$[\text{ACC}] \rightarrow \beta$
加	$\gamma$	$[\text{ACC}] + [\gamma] \rightarrow \text{ACC}$
乘	$\delta$	$[\text{ACC}] \times [\delta] \rightarrow \text{ACC}$
打印	$\sigma$	$[\sigma] \rightarrow \text{打印机}$
停机		

5

计算  $ax^2 + bx + c$  程序清单

指令和数据存于主存单元的地址	指令		注释
	操作码	地址码	
0	000001	0000001000	取数 $x$ 至ACC
1	000100	0000001001	乘 $a$ 得 $ax$ ,存于ACC中
2	000011	0000001010	加 $b$ 得 $ax+b$ ,存于ACC中
3	000100	0000001000	乘 $x$ 得 $(ax+b)x$ ,存于ACC中
4	000011	0000001011	加 $c$ 得 $ax^2 + bx + c$ ,存于ACC
5	000010	0000001100	将 $ax^2 + bx + c$ ,存于主存单元
6	000101	0000001100	打印
7	000110		停机
8	$x$		原始数据 $x$
9	$a$		原始数据 $a$
10	$b$		原始数据 $b$
11	$c$		原始数据 $c$
12			存放结果

## 2. 计算机的解题过程



### (1) 存储器的基本组成



存储体 – 存储单元 – 存储元件 (0/1)

大楼 – 房间 – 床位 (无人/有人)

存储单元 存放一串二进制代码

存储字 存储单元中二进制代码的组合

存储字长 存储单元中二进制代码的位数

每个存储单元赋予一个地址号

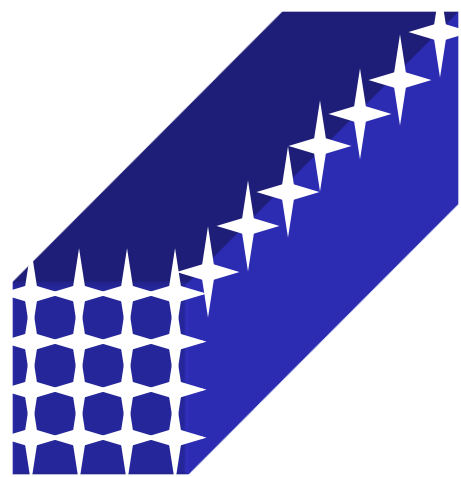
按地址寻访

# (1) 存储器的基本组成



**MAR** 存储器地址寄存器  
反映存储单元的个数

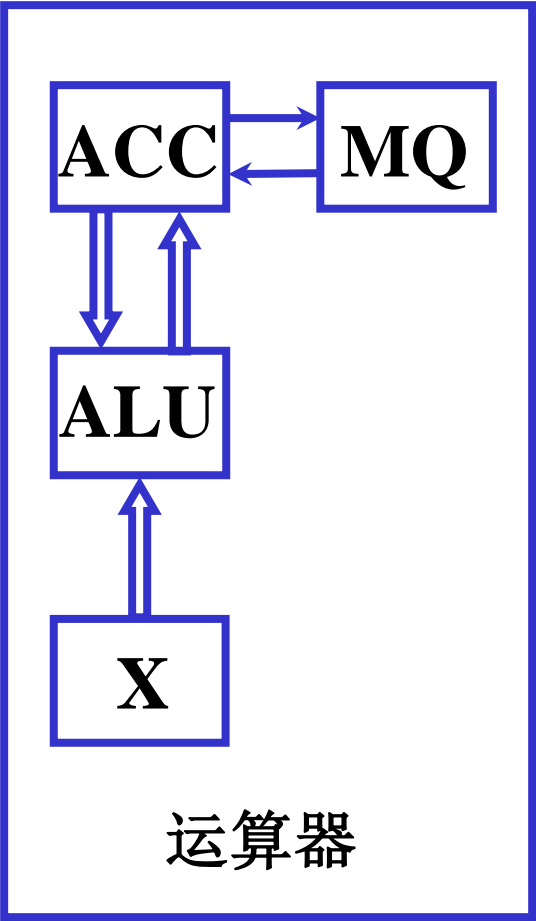
**MBR** 存储器数据寄存器  
反映存储字长



设  $MAR = 4$  位  
 $MBR = 8$  位  
存储单元个数 16  
存储字长 8



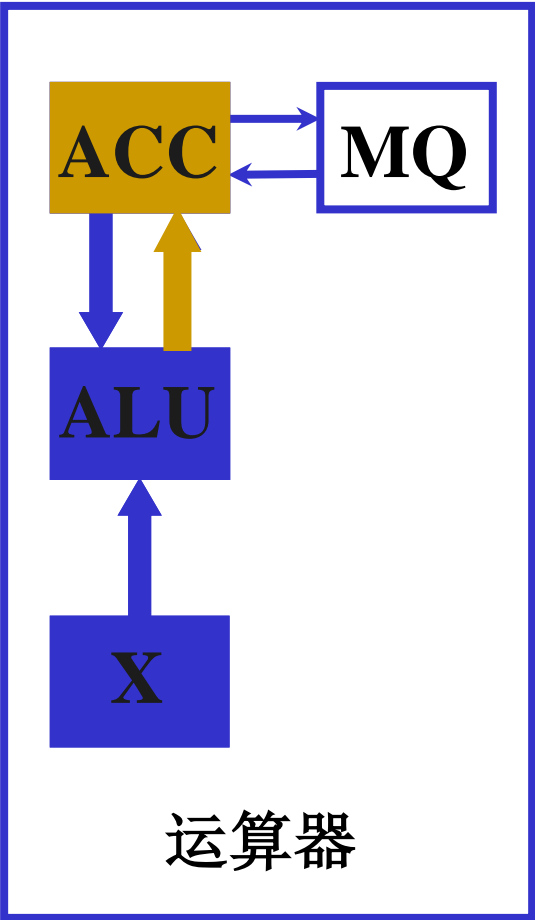
## (2)运算器的基本组成及操作过程



	ACC	MQ	X
加法	被加数 和		加数
减法	被减数 差		减数
乘法	乘积高位	乘数 乘积低位	被乘数
除法	被除数 余数	商	除数



# ① 加法操作过程



指令

加

M

初态

ACC

被加数

[M]

→

X

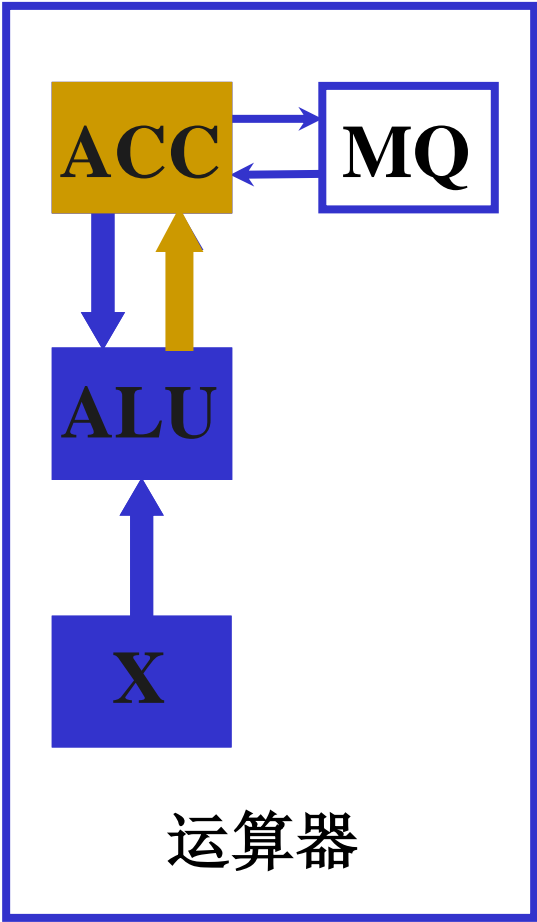
[ACC]+[X]

→

ACC

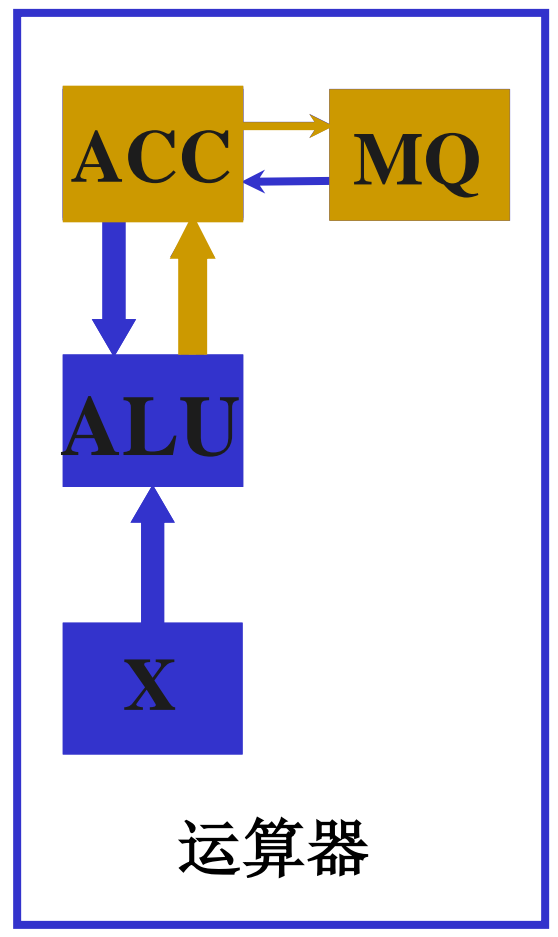


## ② 减法操作过程



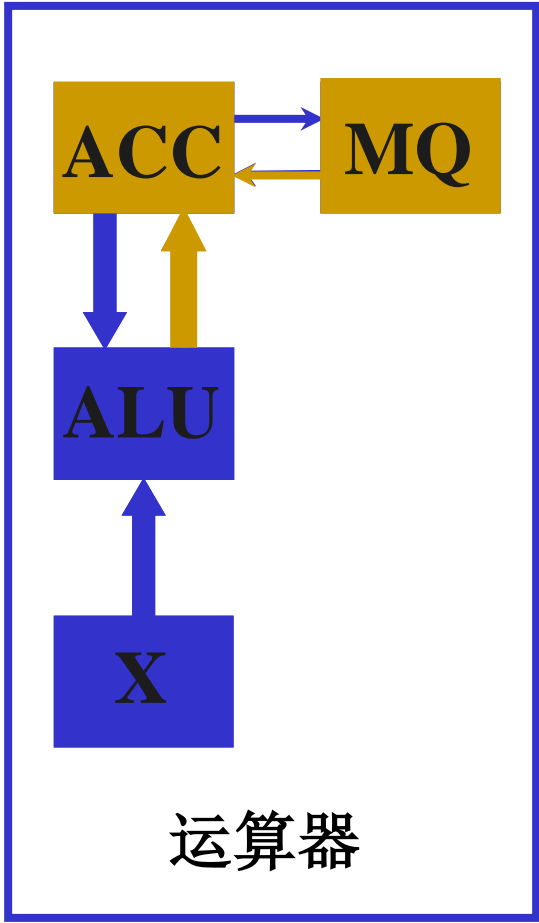
指令	减	M
初态	ACC	被减数
	$[M] \longrightarrow$	X
	$[ACC] - [X] \longrightarrow$	ACC

③ 乘法操作过程



指令	乘	M
初态	ACC	被乘数
	[M] →	MQ
	[ACC] →	X
	0 →	ACC
	[X] × [MQ] → ACC // MQ	

# ④ 除法操作过程



指令	除	M
----	---	---

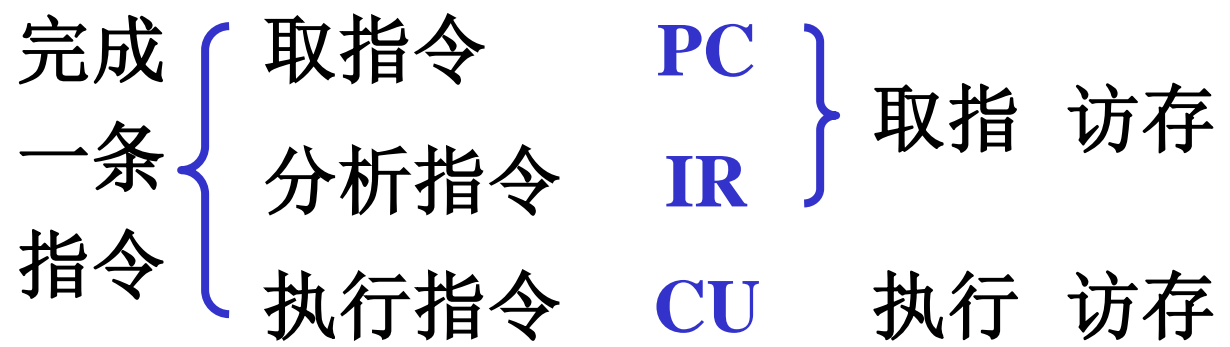
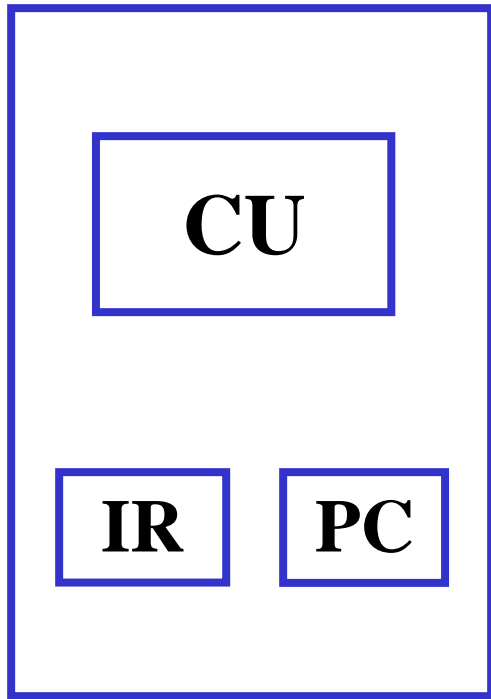
初态      ACC      被除数

$[M] \longrightarrow X$

$[ACC] \div [X] \longrightarrow MQ$

余数在ACC中

### (3) 控制器的基本组成



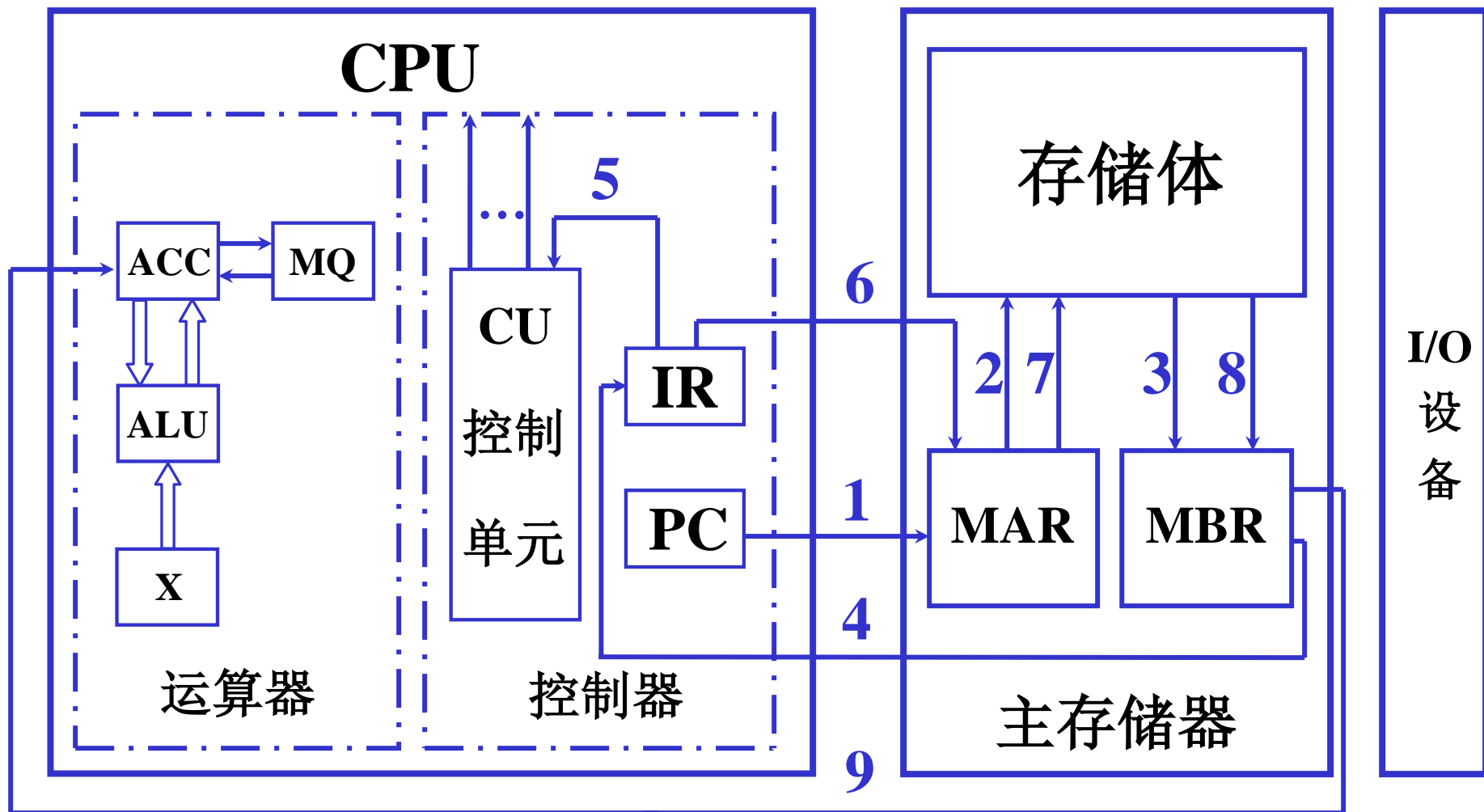
**PC** 存放当前欲执行指令的地址，  
具有计数功能  $(PC) + 1 \rightarrow PC$

**IR** 存放当前欲执行的指令



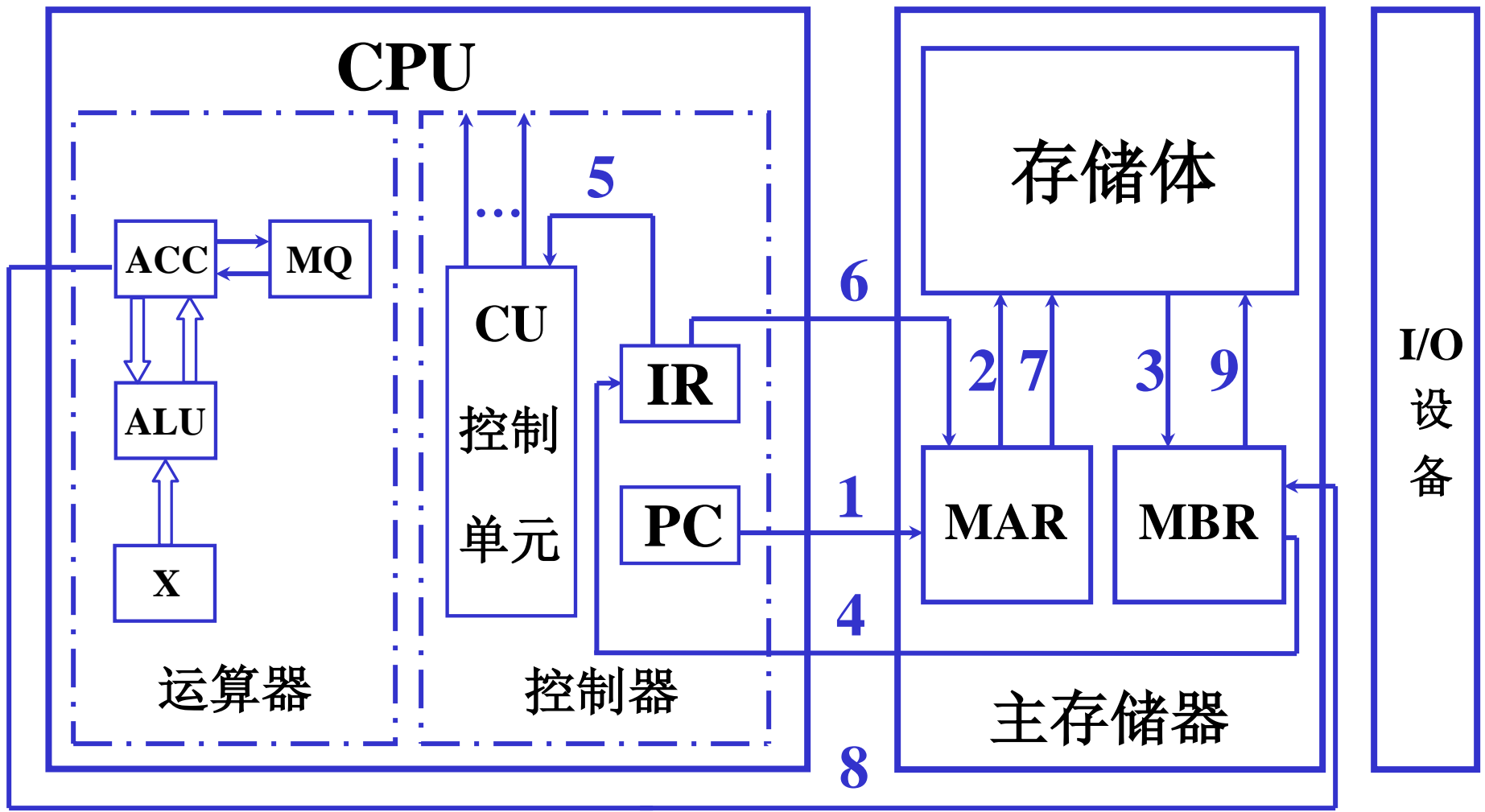
# (4) 主机完成一条指令的过程

## 以取数指令为例



# (4) 主机完成一条指令的过程

以存数指令为例






## (5) $ax^2 + bx + c$ 程序的运行过程

- 将程序通过输入设备送至计算机
- 程序首地址  $\longrightarrow$  PC
- 启动程序运行
- 取指令  $PC \rightarrow MAR \rightarrow M \rightarrow MBR \rightarrow IR, (PC)+1 \rightarrow PC$
- 分析指令  $OP(IR) \rightarrow CU$
- 执行指令  $Ad(IR) \rightarrow MAR \rightarrow M \rightarrow MBR \rightarrow ACC$
- $\vdots$
- 打印结果
- 停机

# 思考题

 下列选项中，描述浮点数操作速度指标的是（）。

A. MIPS B. CPI C. IPC D. MFLOPS

 假定计算机M1和M2具有相同的指令集体系结构(ISA)，主频分别为1.5GHz和1.2GHz。在M1和M2上运行某基准程序P，平均CPI分别为2和1，则程序P在M1和M2上运行时间的比值是（）。

A. 0.4 B. 0.625 C. 1.6 D. 2.5

# 本章小结

## 1. 计算机的发展（了解）

冯诺依曼计算机的特点（掌握）

## 2. 计算机系统的基本组成（掌握）

系统组成、硬件组成、五大部件原理和功能

## 3. 计算机系统的层次结构（理解）

## 4. 计算机的性能指标（掌握）

含义、计算

## 5. 计算机的工作过程（理解）