

# 计算机组成原理

## PRINCIPLES OF COMPUTER ORGANIZATION

主讲教师：隋秀峰

中国科学院大学

中国科学院计算技术研究所

# 计算机革命

- 计算机技术的进步
  - 摩尔定律的驱动
- “计算机科学幻想” 到现实
  - 车载计算机
  - 手机
  - 人类基因组项目
  - WWW
  - 搜索引擎

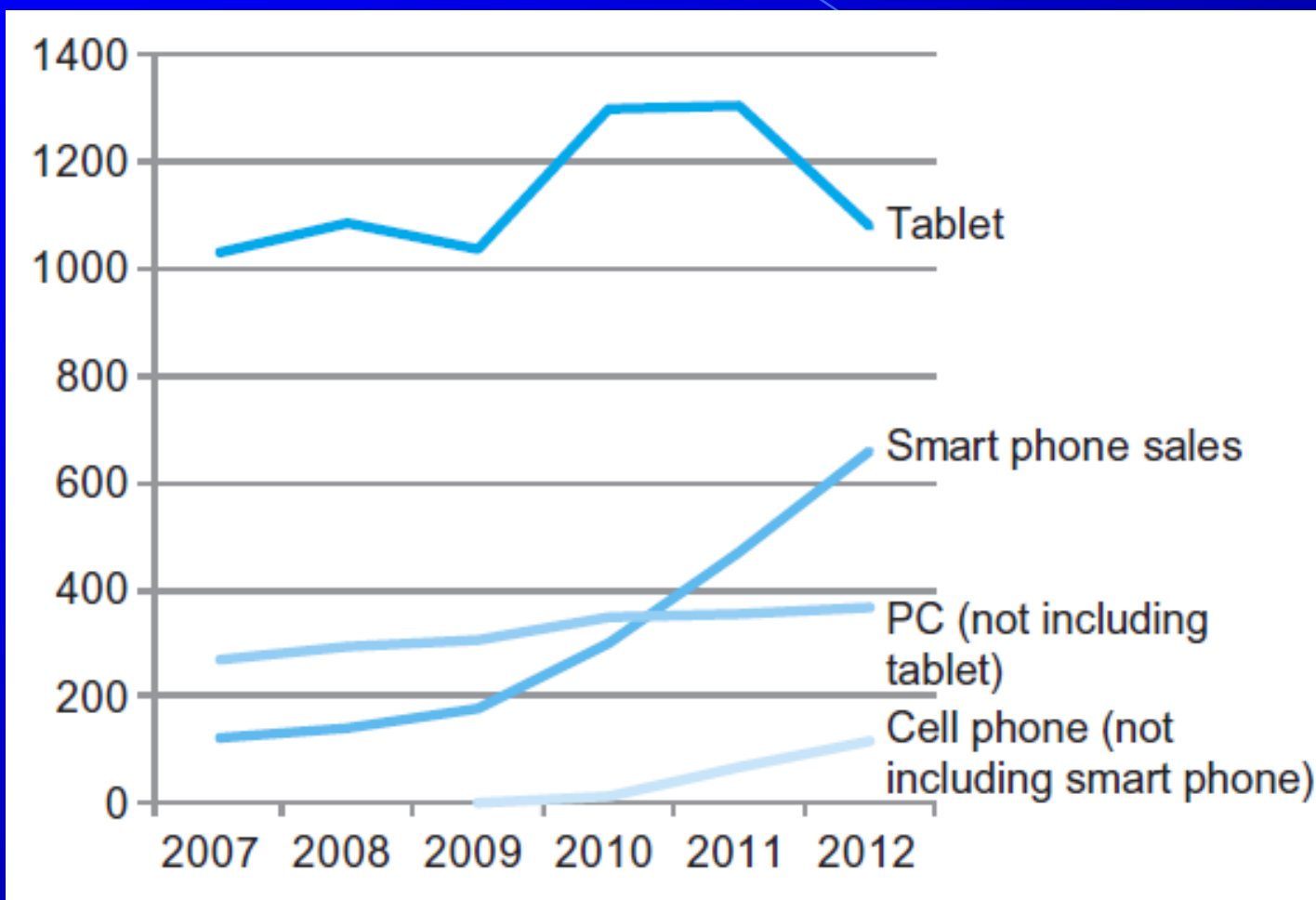
计算无所不在，计算机无所不能



# 计算机的分类

- 个人计算机（PC）
  - 通用，运行第三方软件
  - 追求性能功耗比
- 服务器
  - 基于网络
  - 高性能、高容量、高可靠
  - 功能和价格有很大的伸缩范围
- 超级计算机
  - 用于高端科学和工程计算
  - 最高的计算能力，但只占服务器很小一部分
- 嵌入式计算机
  - 通常和硬件集成在一起以单一系统的方式存在
  - 对功耗、性能、成本有严格限制
  - 运行预定义的一个或者一组应用程序

# 后PC时代

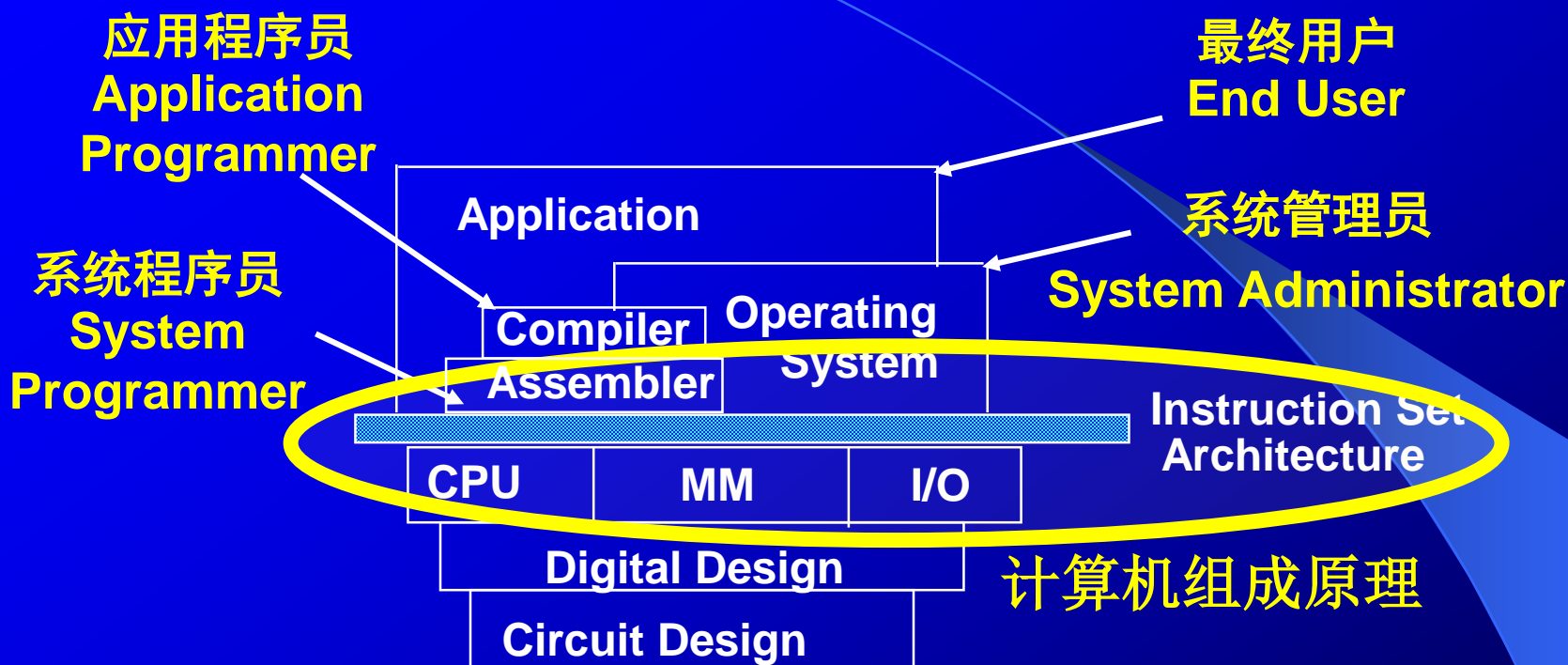


# 后PC时代

- 个人移动设备 (PMD)
  - 由电池供电
  - 通过无线方式连接到网络
  - 价格通常只有几百美元
  - 智能手机、平板电脑
- 云计算
  - Ware-house Scale Computer (WSC)
  - 实现软件即服务 (SaaS)
  - PMD和云上各运行应用的一部分
  - Amazon、Google、BAT等



# 课程地位



- 计算机系统的层次结构
  - 指令系统（即ISA）是软/硬件的交界面
- 不同用户工作在不同层次，所看到的计算机不一样
- 本课程主要内容处于最核心的部分！

# 课程目标

- 系统的理解并掌握计算机的运行原理
  - 学习计算机硬件的组成
    - 高级语言程序如何翻译成硬件之间的语言
    - 硬件如何执行程序
  - 理解计算机硬件/软件的协同机制（接口）
  - 影响程序性能的因素、如何改进程序性能（尤其是硬件设计上）

让你成为懂计算机的人



# 课程概况

- 学时分配
  - 理论：60学时；实验：40学时
- 授课教师
  - 主讲教师：隋秀峰（[suixiufeng@ict.ac.cn](mailto:suixiufeng@ict.ac.cn)）
  - 助课教师：姚治成（[yaozhcheng@ict.ac.cn](mailto:yaozhcheng@ict.ac.cn)）、任睿（[renrui@ict.ac.cn](mailto:renrui@ict.ac.cn)）
- 考核方式
  - 期中+期末考试：80%；作业+研讨：20%



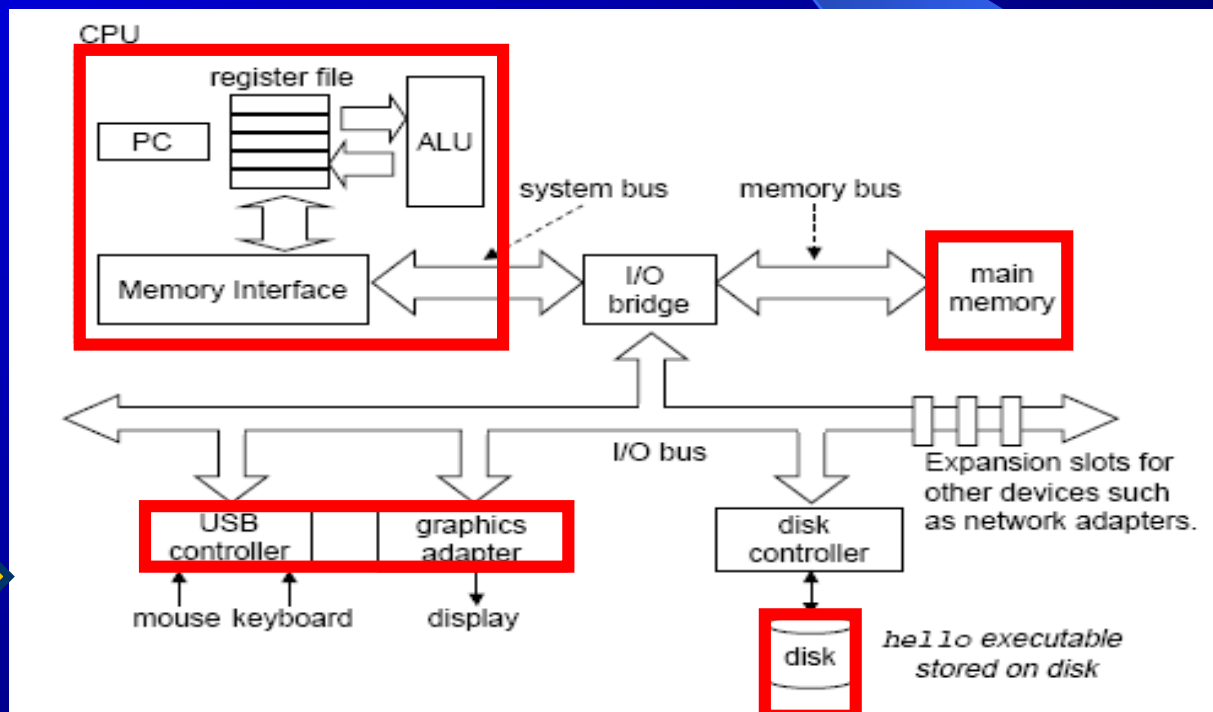
# 课程大纲

- 计算机系统的基本概念
- 计算机中数的表示
- 指令系统
- 数据通路及控制部件
- 存储系统
- 中断与异常
- 输入输出系统

# 计算机系统的基本概念

## ● 解剖计算机

- 计算机软硬概念；计算机系统的层次结构；计算机的基本组成、冯·诺依曼计算机的特点；计算机的硬件框图及工作过程；评价CPU性能的主要技术指标；计算机的发展及应用、计算机基本结构的演变与实现技术



# 计算机中数的表示

- 理解0和1的魔力
- 例子: (float) 17825792 + 1 = 17825793?
  - 二进制表示为0001 0001 0000 0000 0000 0000 0000
  - 二进制浮点表示为 $1.0001 \times 2^{11000}$
  - 根据IEEE 754浮点数标准, 只能表示24位尾数
  - 故(float) 17825792 + 1, 由于计算机表示对精度的影响仍为17825792



# 指令系统

- 计算机的语言

- 例子: MIPS-lite Instruction Subset

- ADDU and SUBU

- `addu rd,rs,rt`
- `subu rd,rs,rt`

- OR Immediate:

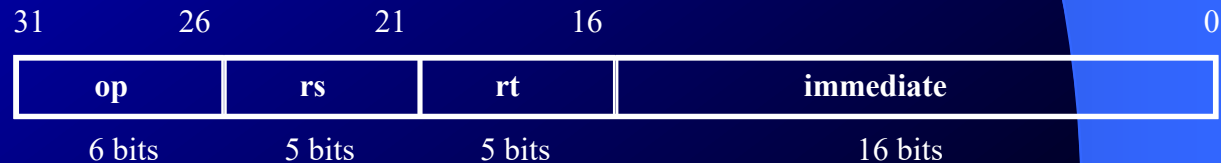
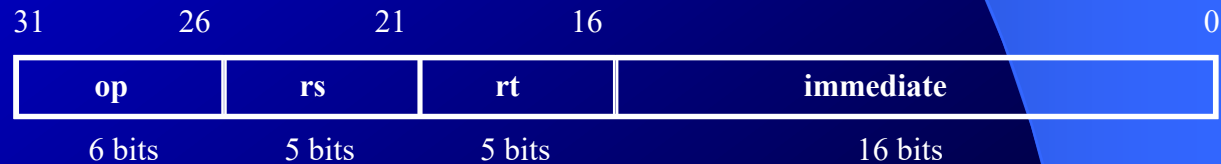
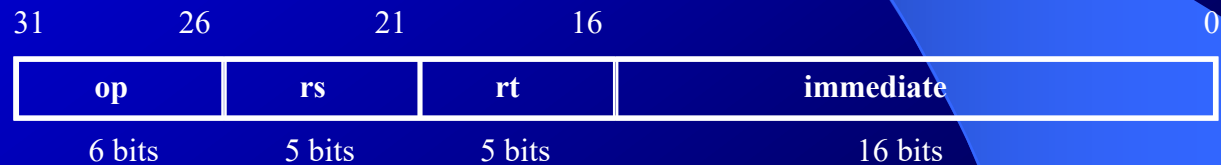
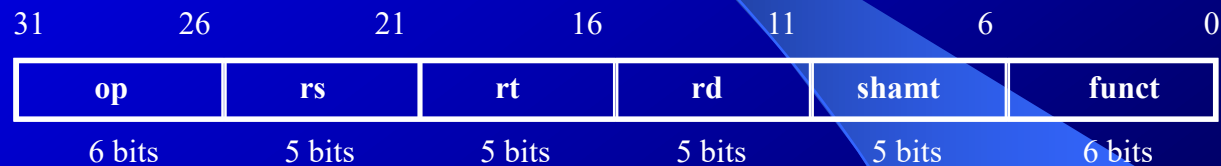
- `ori rt,rs,imm16`

- LOAD and STORE Word

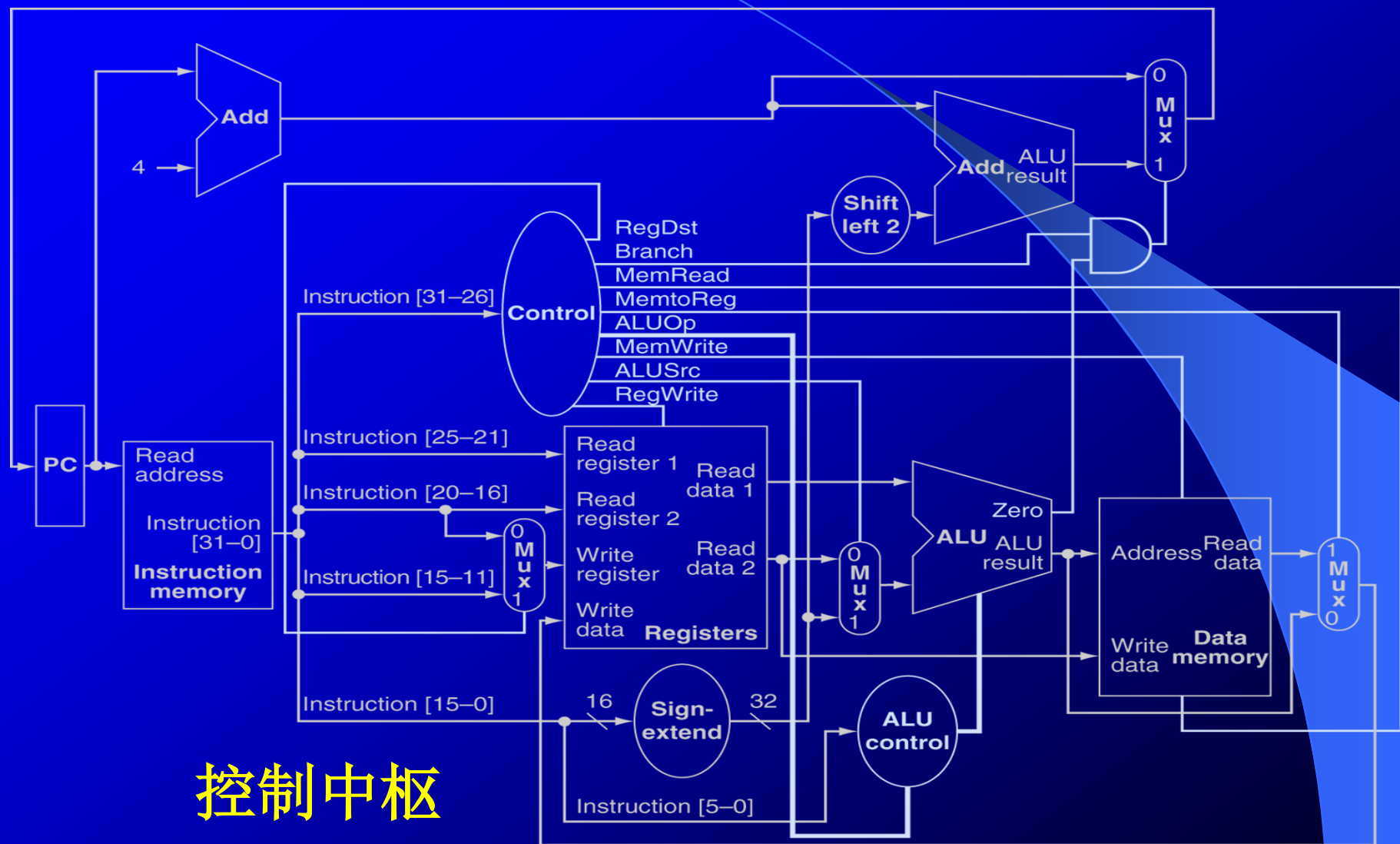
- `lw rt,rs,imm16`
- `sw rt,rs,imm16`

- BRANCH:

- `beq rs,rt,imm16`



# 数据通路及控制部件



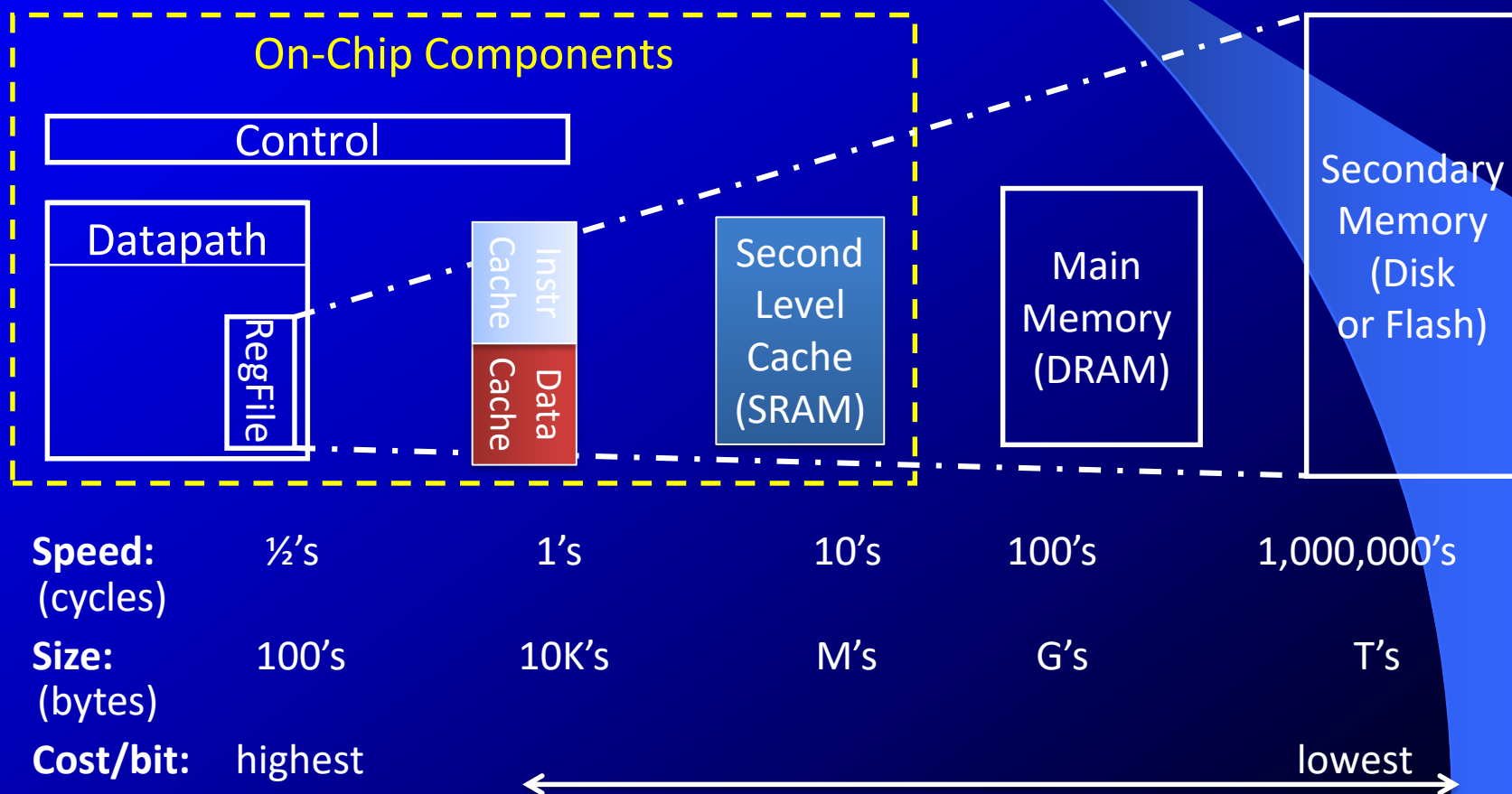
控制中枢



# 存储系统

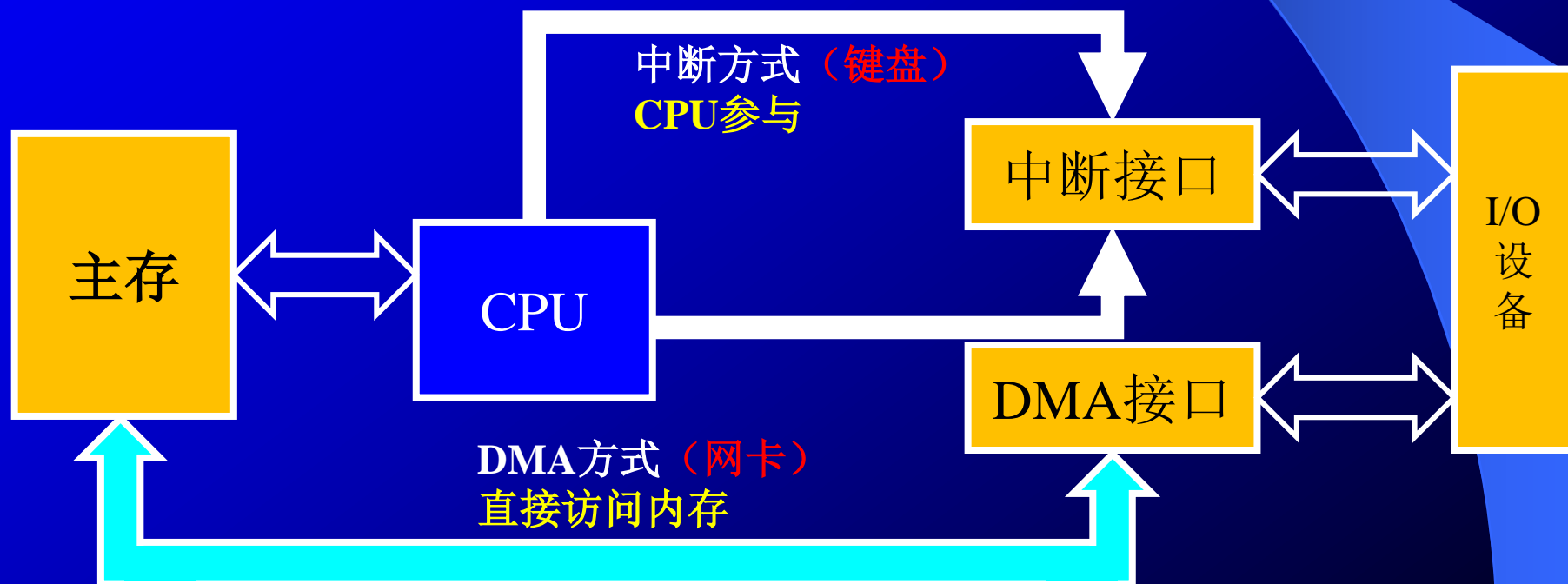
## ● 数据存放和移动规则

- 存储器分类和存储器的层次结构；主存储器；高速缓冲存储器；虚拟存储器



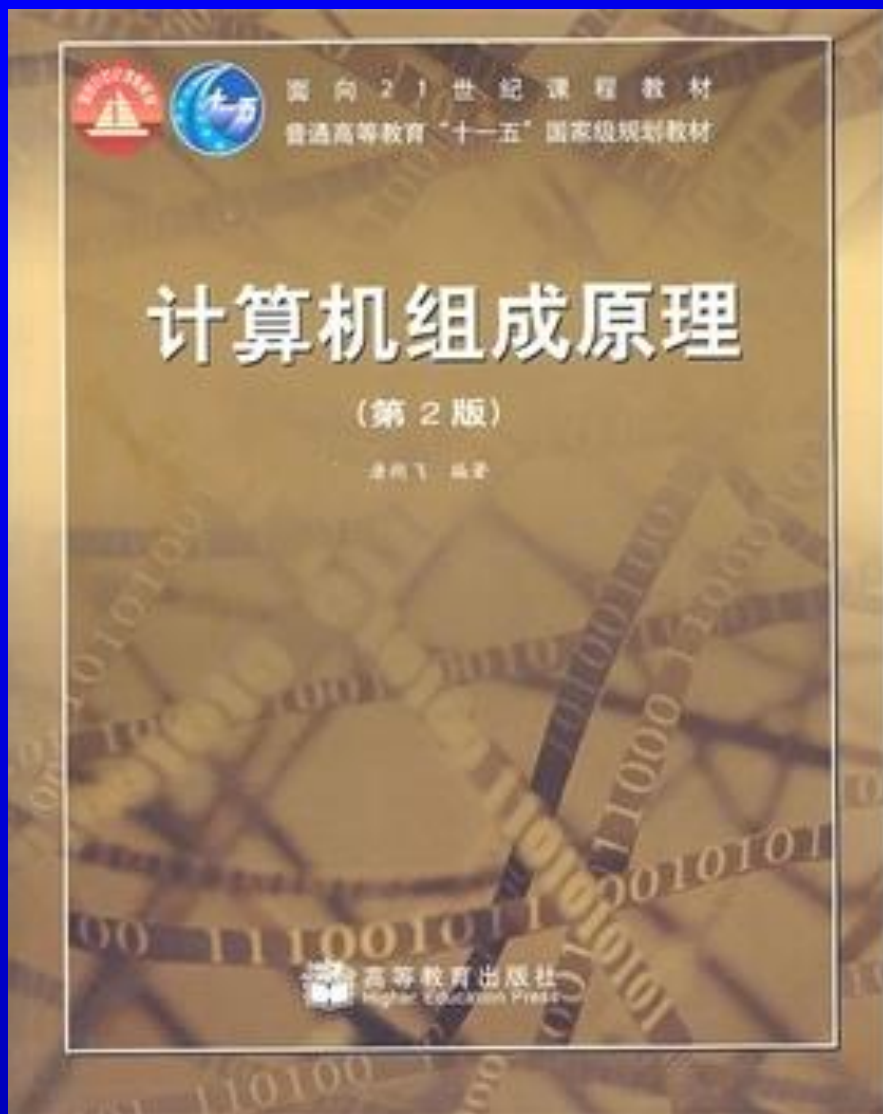
# 输入输出系统

- 例子1：打字-键盘中断
- 例子2：网络上下载文件-网卡DMA





# 教材和参考书





# 这门课程

- **强度大**：学习基本原理、硬件描述语言、软件工具、硬件调试、芯片手册
- 态度决定一切（ATTITUDE）
- 奠定你对计算机的认识，铸就你未来成就的基石
- 期待着你最终能感受到“一览众山小”的**巅峰体验**



# 第1讲 计算机系统概论

## 1.1 计算机系统简介

## 1.2 计算机的基本组成

## 1.3 计算机硬件的主要技术指标

## 1.4 课程结构



# 1.1 计算机系统简介

## 一、计算机的软硬件概念

### 1. 计算机系统

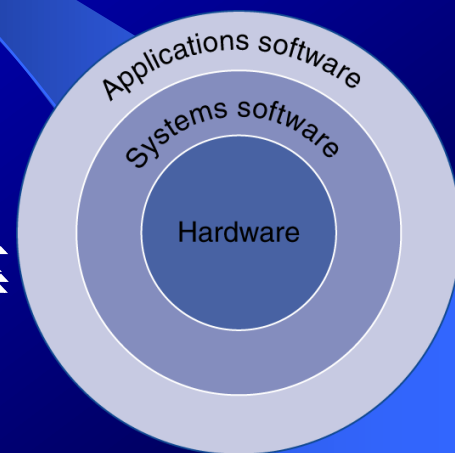
计算机系统

硬件

计算机的实体，  
如主机、外设等

软件

由具有各类特殊功能  
的信息（程序）组成



## 软件

**系统软件** 用来管理整个计算机系统

语言处理程序

操作系统 (处理输入输出; 管理内存和存储; 调度任务、共享资源)

服务性程序

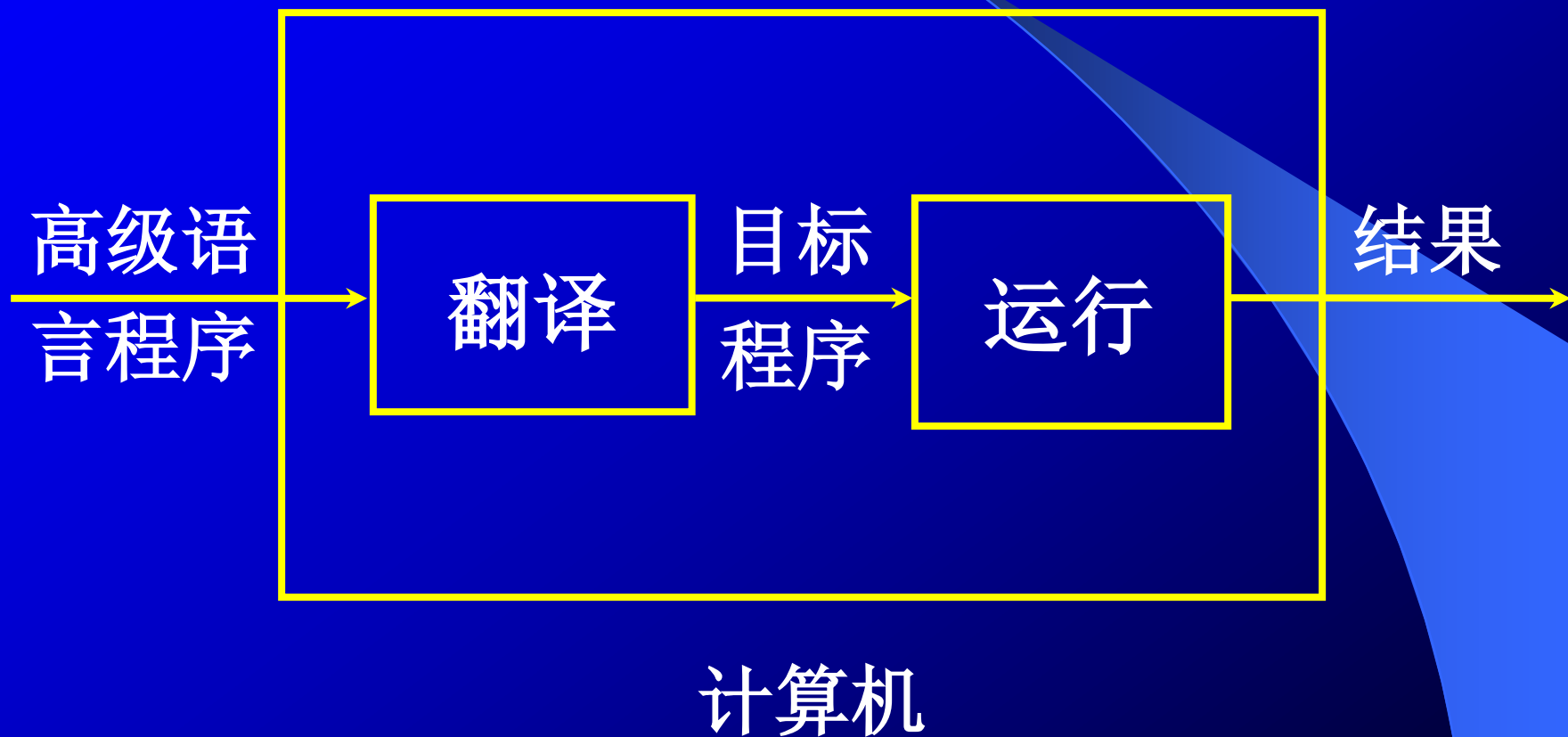
数据库管理系统

网络软件

**应用软件** 按任务需要编制成的各种程序  
(高级语言编制)

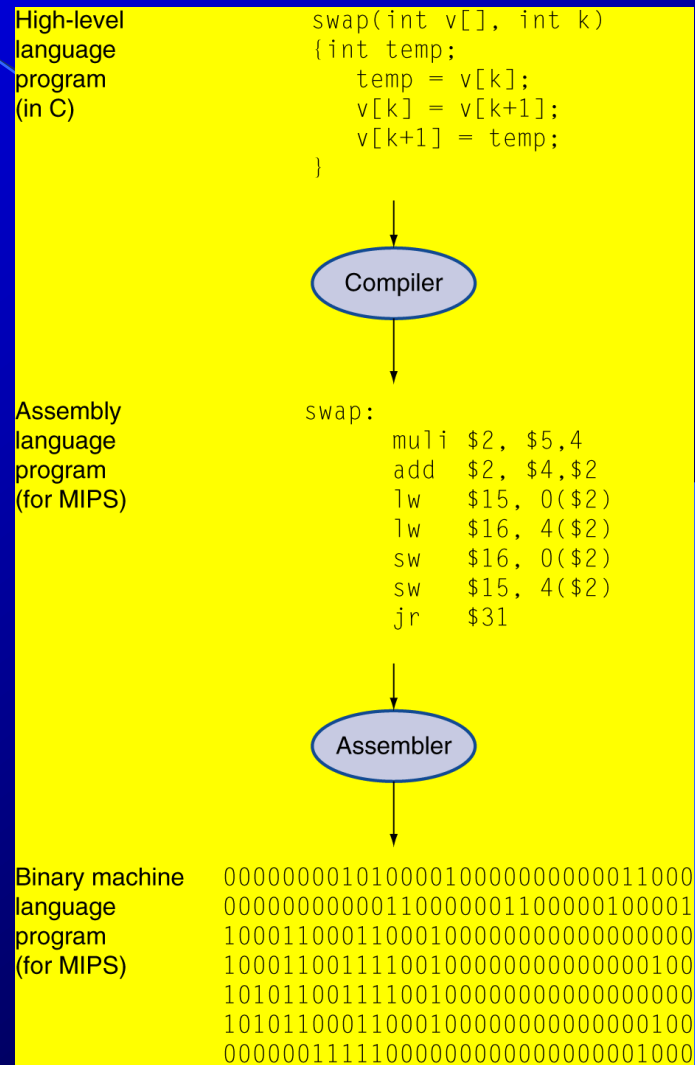


## 2. 计算机的解题过程

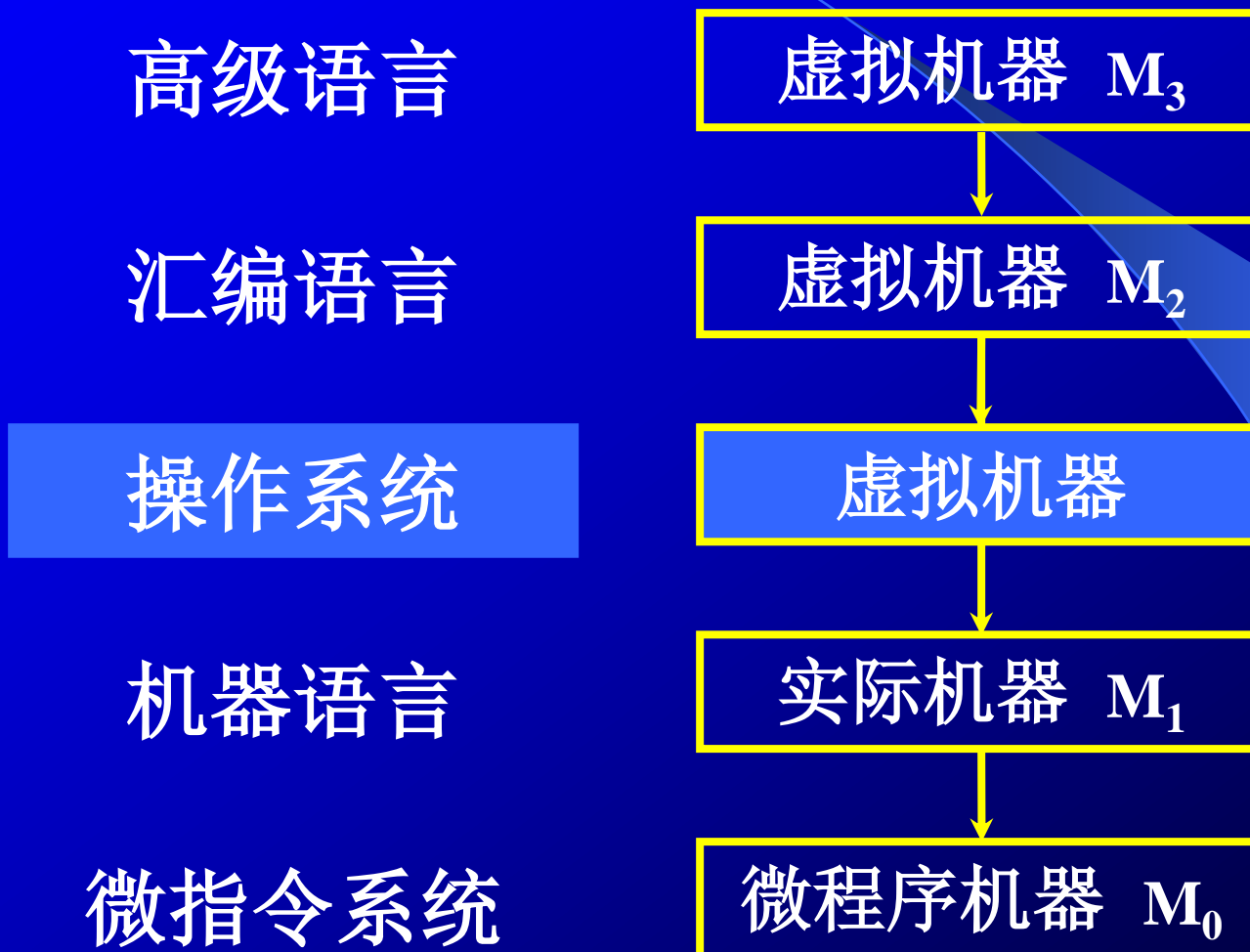


### 3、计算机的语言

- 机器语言
  - 二进制(位)
  - 指令和数据编码
- 汇编语言
  - 机器指令的最直接表示
- 高级语言
  - 与问题领域最接近的抽象层次
  - 效率和可移植性



## 二、计算机系统的层次结构



软件

虚拟机器  $M_4$ 

用编译程序翻译成汇编语言程序

虚拟机器  $M_3$ 

用汇编程序翻译成机器语言程序

虚拟机器  $M_2$ 

用机器语言解释操作系统

硬件

实际机器  $M_1$ 

用微指令解释机器指令

微程序机器  $M_0$ 

由硬件直接执行微指令



# 三、计算机体系结构和计算机组成

1.1

## 计算机 体系结构

有无乘法指令

程序员所见到的计算机系统的属性  
概念性的结构与功能特性

（指令系统、数据类型、寻址技术、I/O机理）

## 计算机 组成

实现计算机体系结构所体现的属性  
（具体指令的实现）

如何实现乘法指令



# 1.2 计算机的基本组成

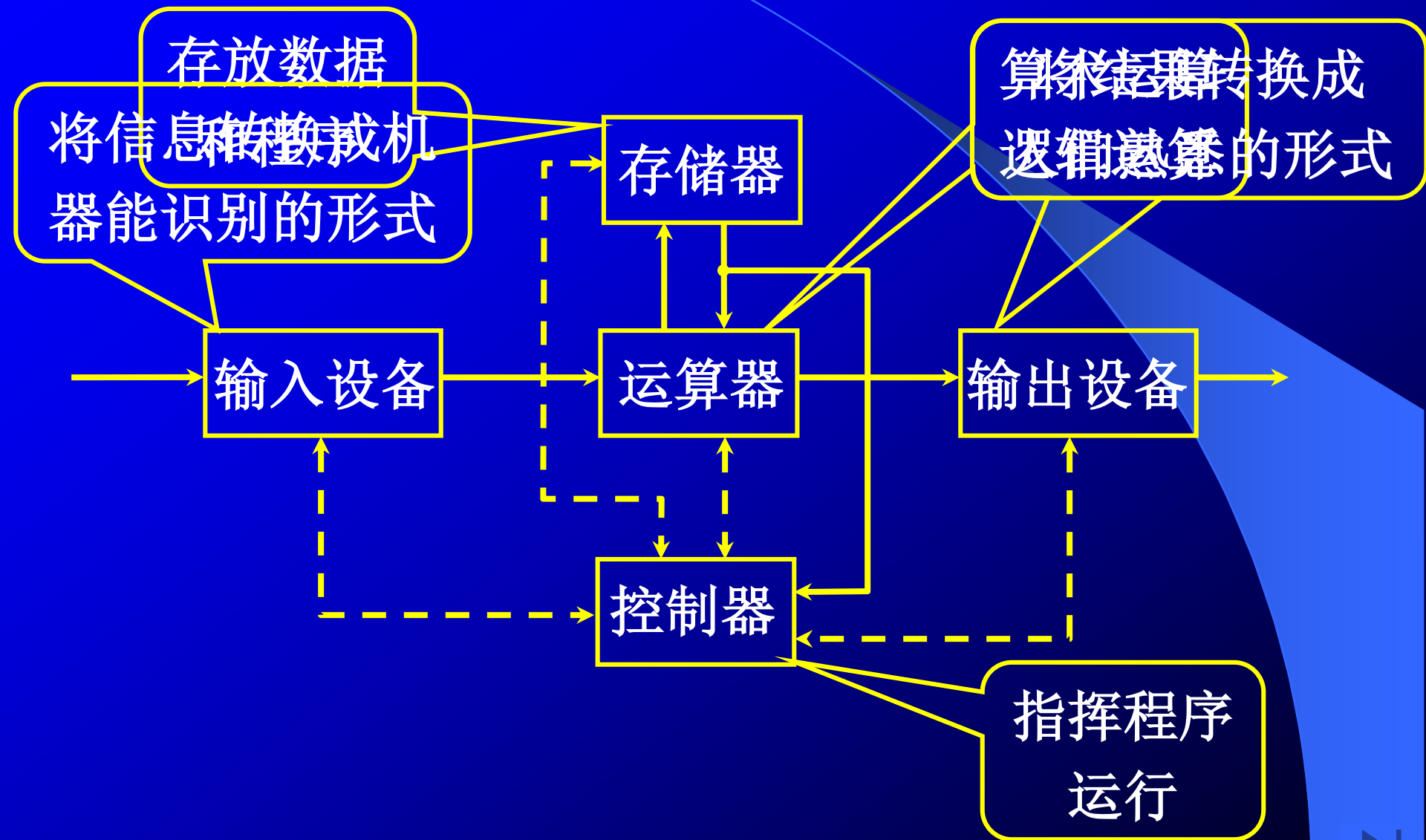
## 一、冯·诺依曼计算机的特点

1. 计算机由五大部件组成
2. 指令和数据以同等地位存于存储器，可按地址寻访
3. 指令和数据用二进制表示
4. 指令由操作码和地址码组成
5. 存储程序
6. 以运算器为中心



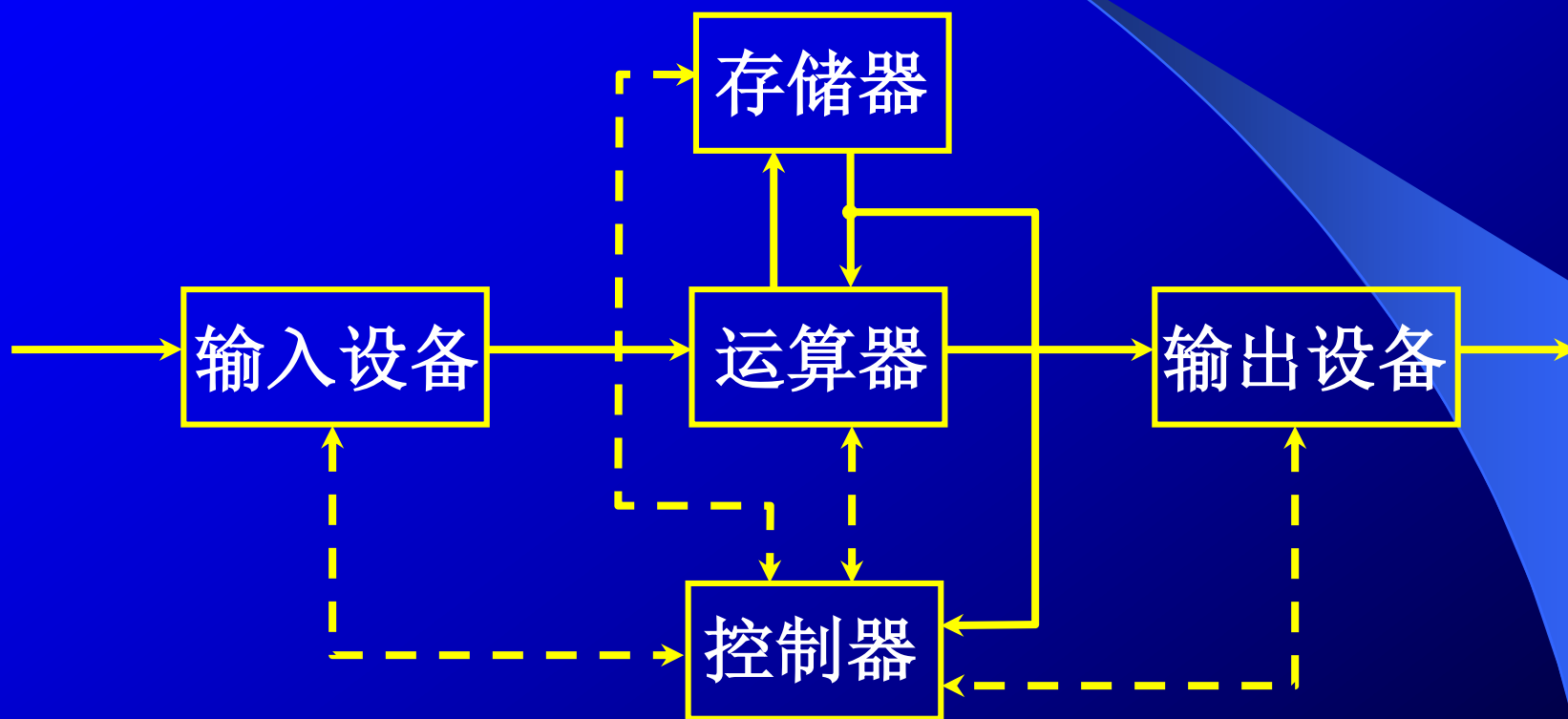
# 冯·诺依曼计算机硬件框图

1.2



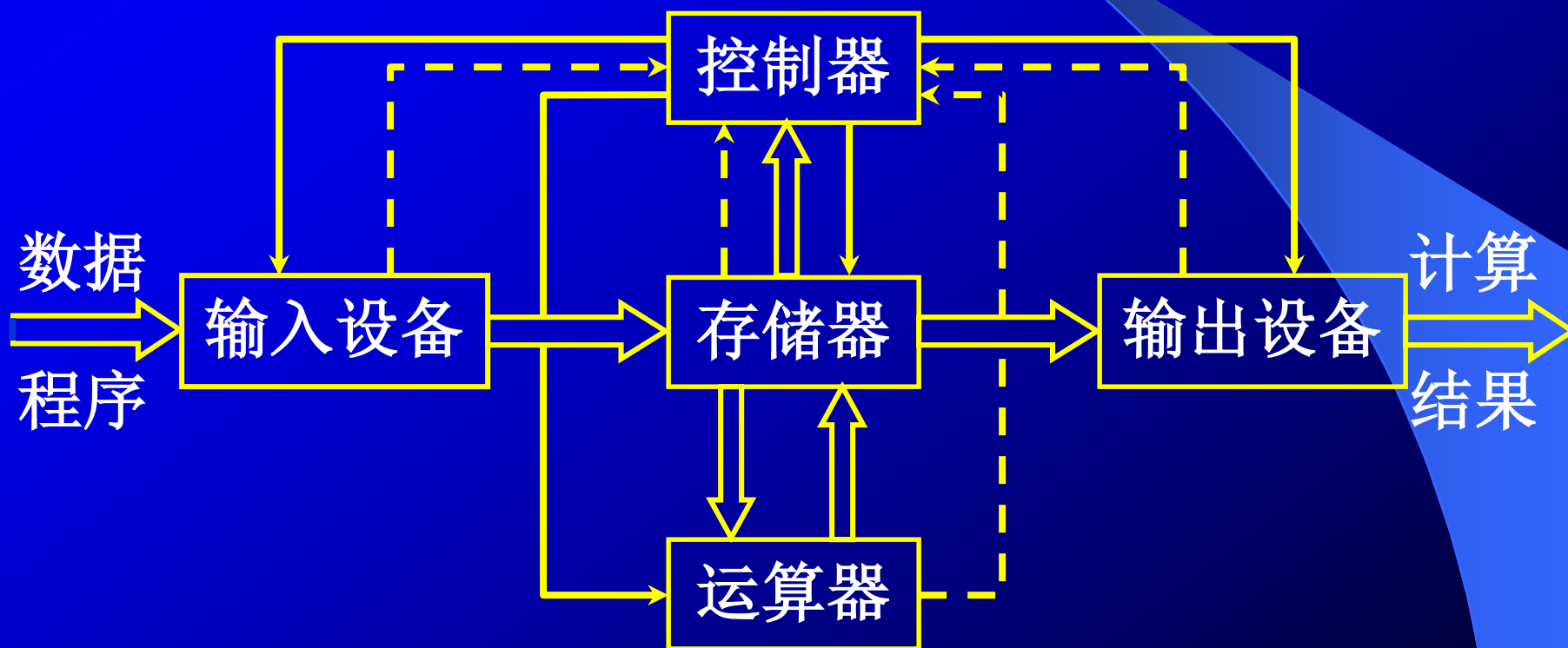
# 冯·诺依曼计算机硬件框图

1.2

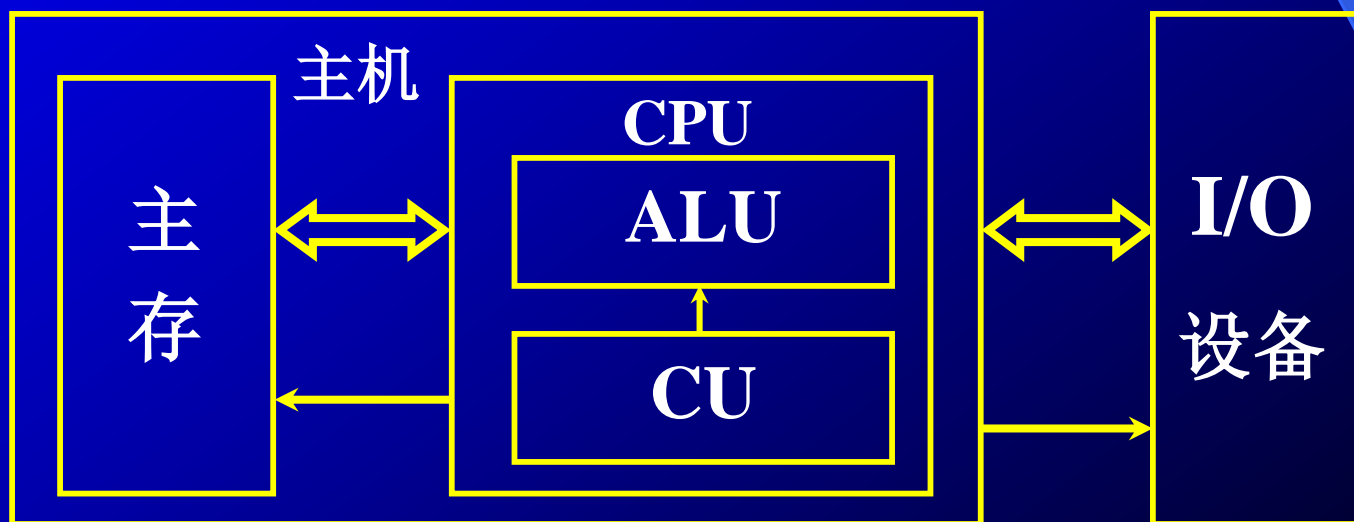
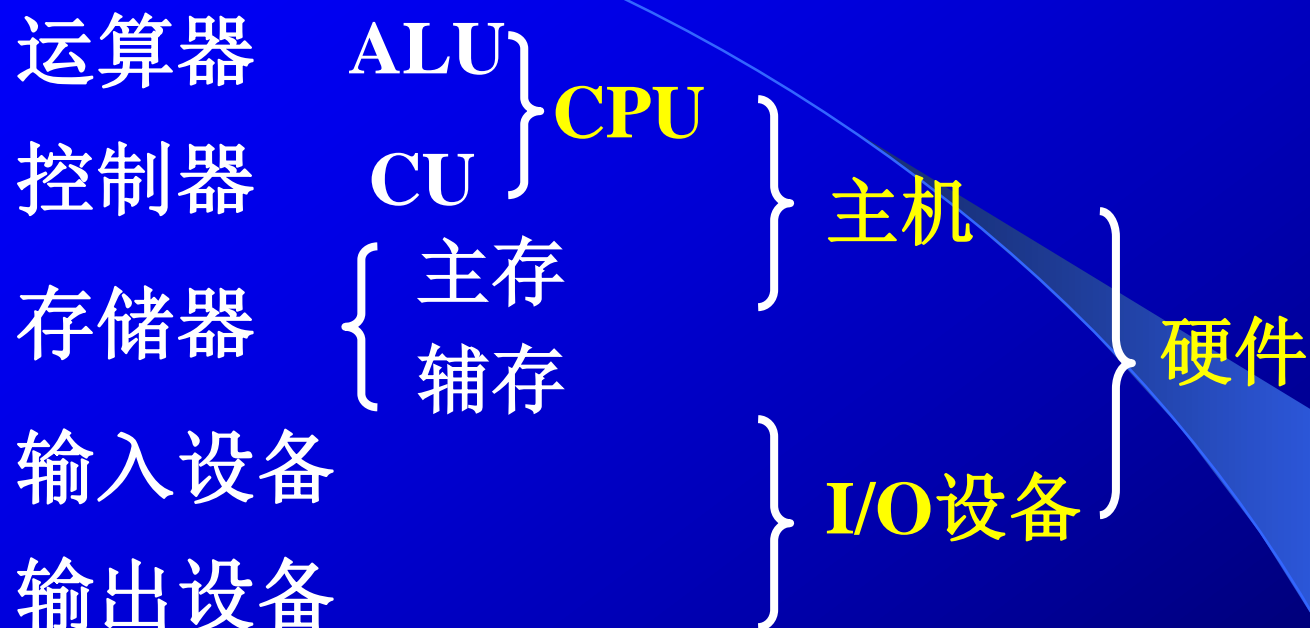


## 二、计算机硬件框图

### 1. 以存储器为中心的计算机硬件框图



## 2. 现代计算机硬件框图



# 三、计算机的工作步骤

## 1.2

### 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

## 1.2

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

# 计算 $ax^2 + bx + c$ 程序清单

## 1.2

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. 计算机的工作过程

## 1.2

### (1) 主存储器的基本组成



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**



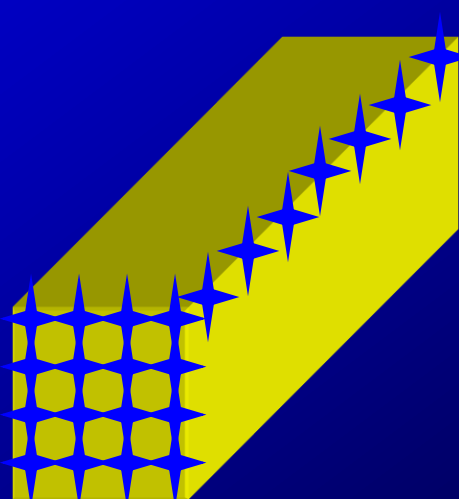
# (1) 存储器的基本组成



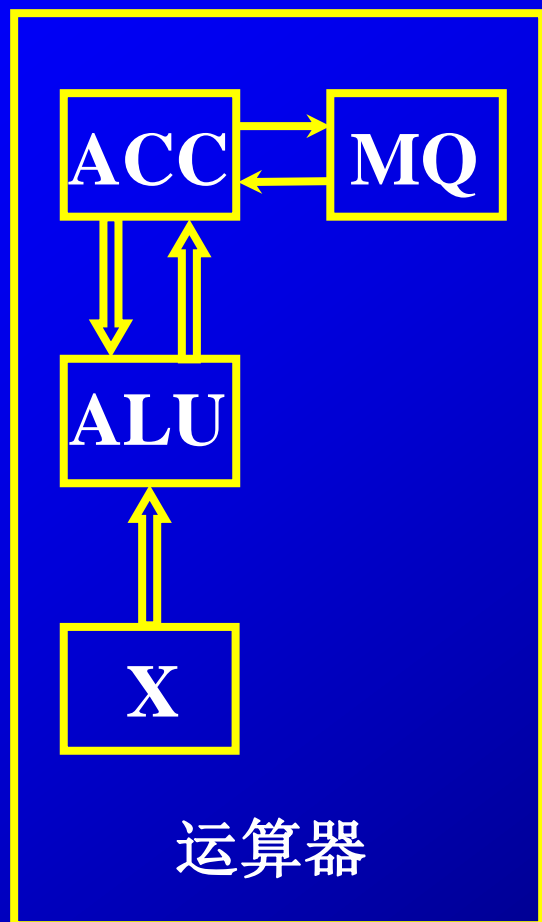
**MAR** 存储器地址寄存器  
反映存储单元的个数

**MDR** 存储器数据寄存器  
反映存储字长

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



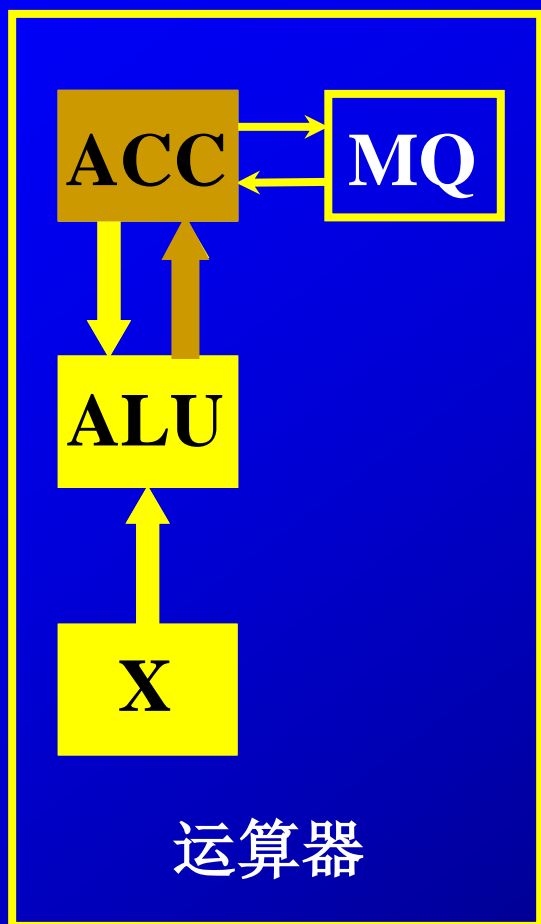
## (2)运算器的基本组成及操作过程



	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

ACC: Accumulator; MQ: Multiplier-Quotient Register

# ① 加法操作过程



指令

加

M

初态

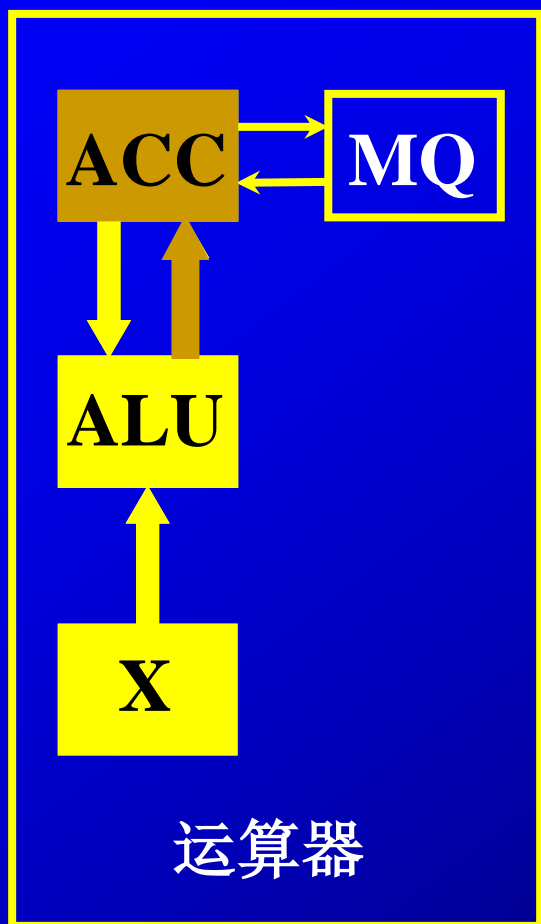
ACC

被加数

$[M] \longrightarrow X$

$[ACC] + [X] \longrightarrow ACC$

## ② 减法操作过程



指令

减

M

初态

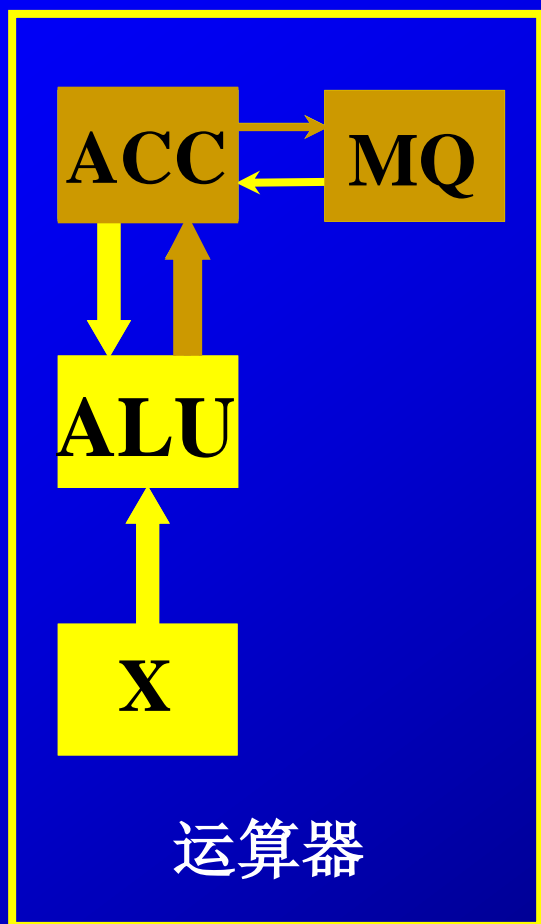
ACC

被减数

$[M] \longrightarrow X$

$[ACC] - [X] \longrightarrow ACC$

## ③ 乘法操作过程



指令

乘

M

初态

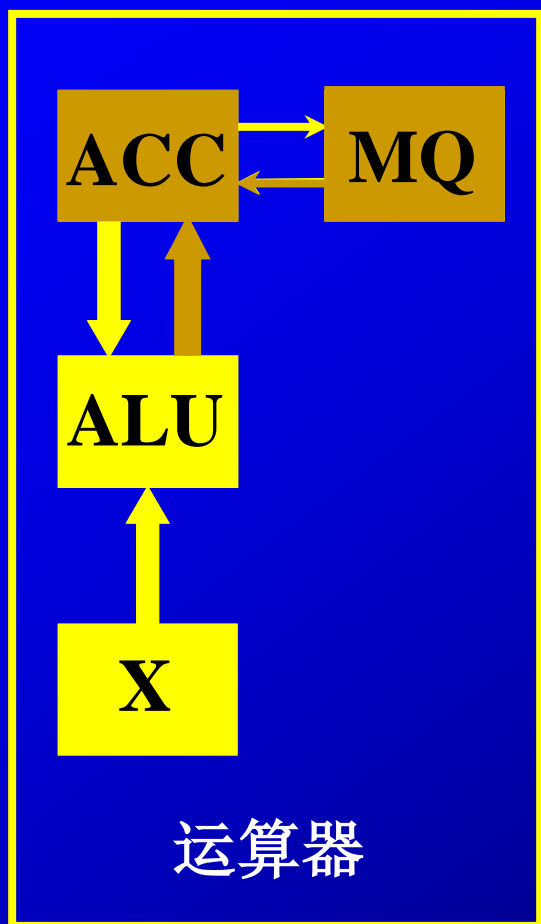
ACC

被乘数

[M]  $\longrightarrow$  MQ[ACC]  $\longrightarrow$  X0  $\longrightarrow$  ACC[X]  $\times$  [MQ]  $\longrightarrow$  ACC // MQ



## ④ 除法操作过程



指令

除

M

初态

ACC

被除数

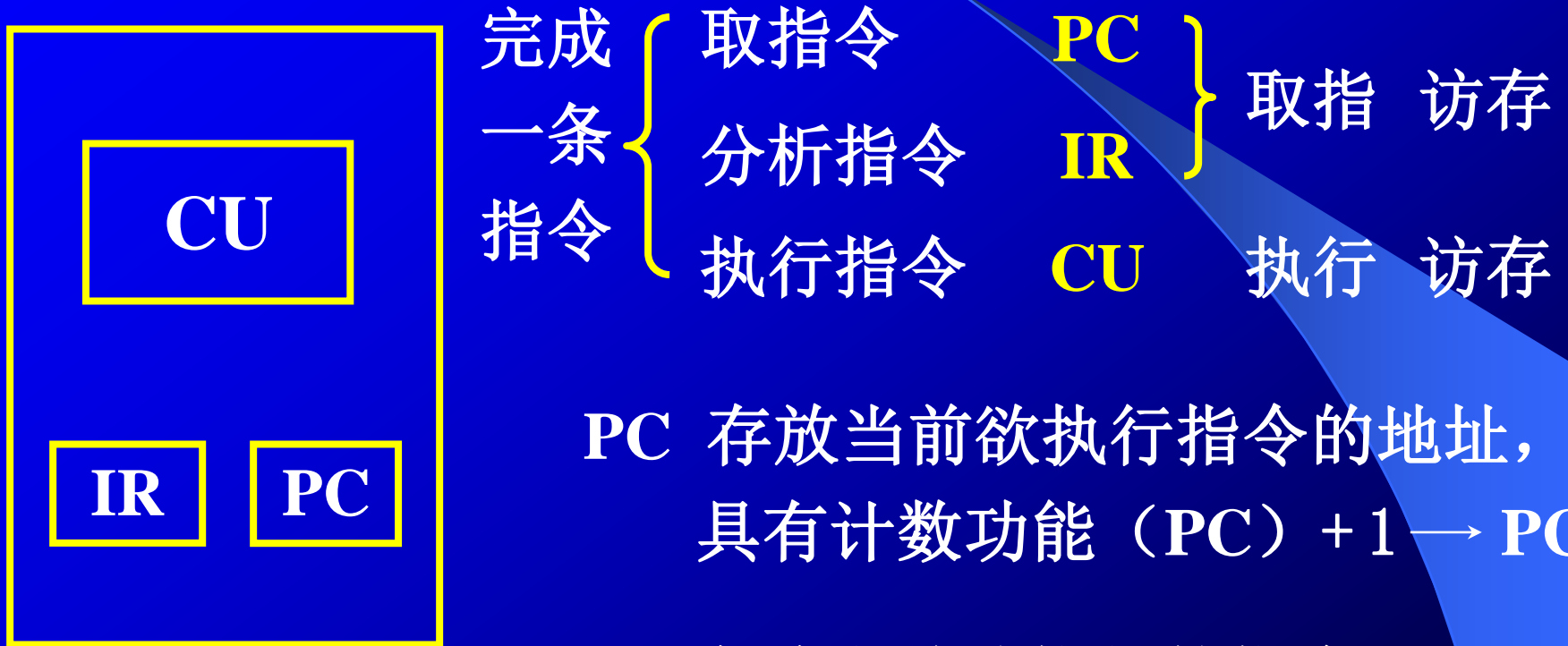
$[M] \longrightarrow X$

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

余数在ACC中

### (3) 控制器的基本组成

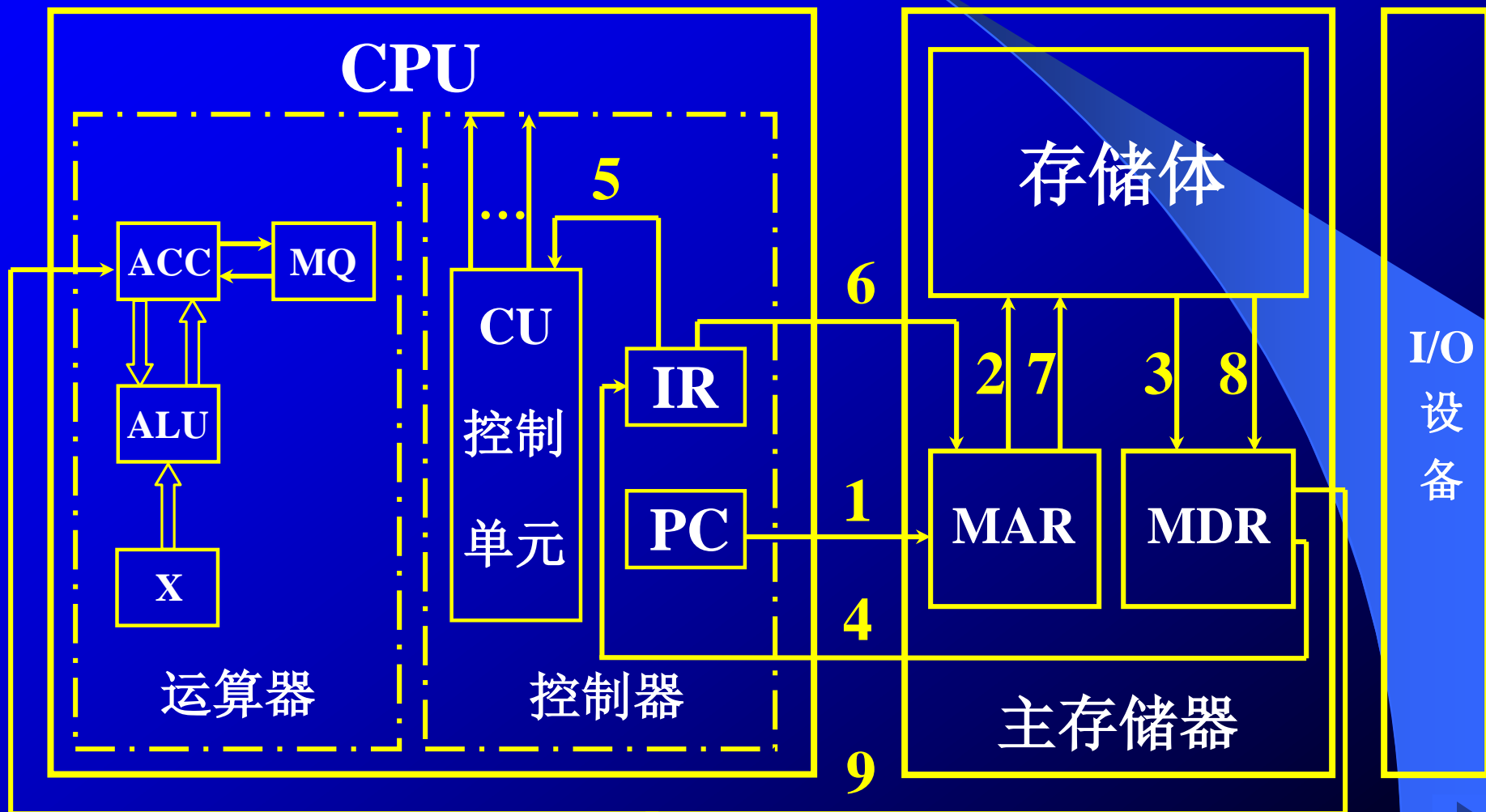
1.2



# (4) 主机完成一条指令的过程

## 以取数指令为例

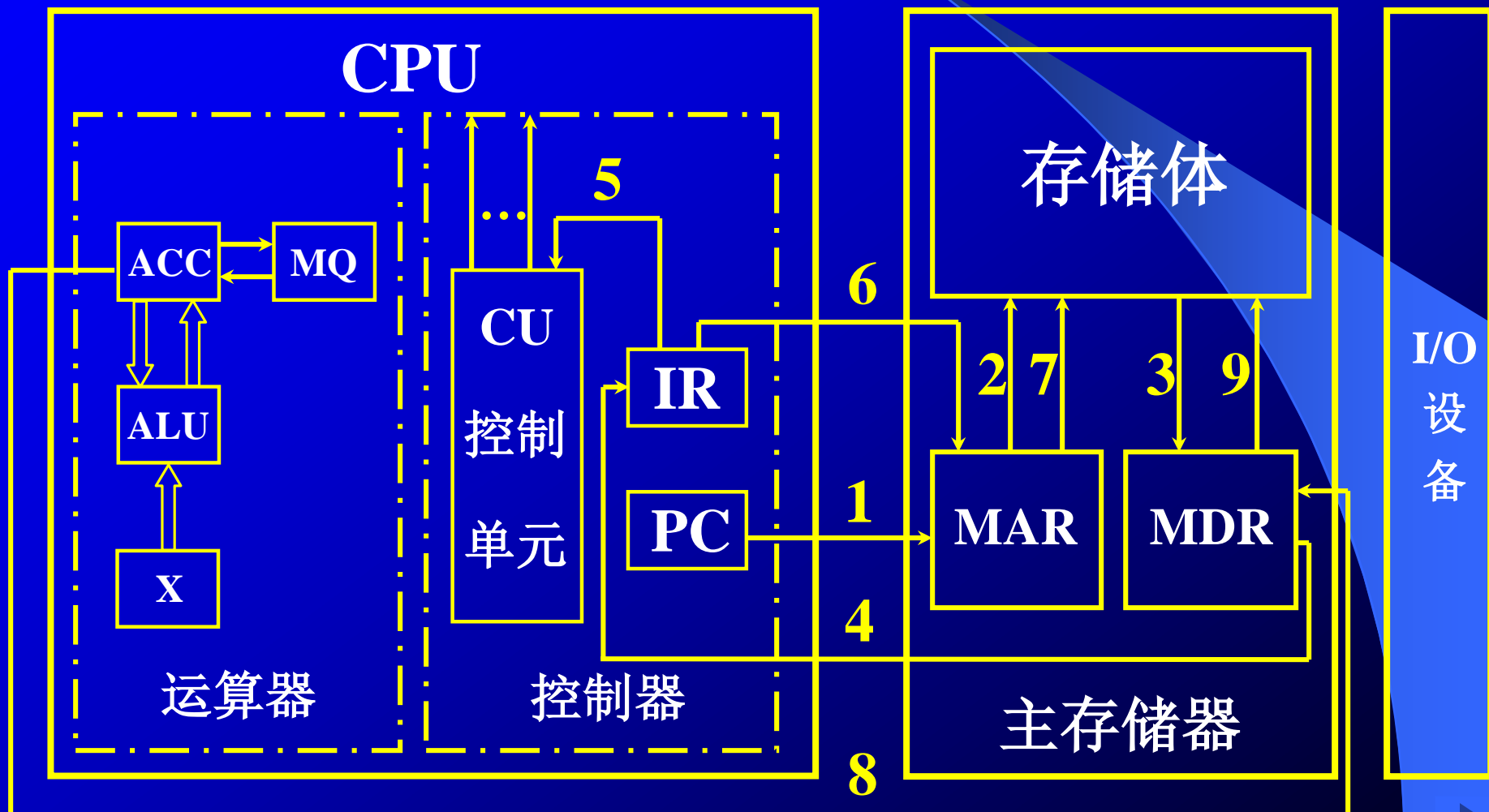
1.2



# (4) 主机完成一条指令的过程

## 以存数指令为例

1.2



## (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  $PC \rightarrow MAR \rightarrow M \rightarrow MDR \rightarrow IR$  ,  $(PC) + 1 \rightarrow PC$
- 分析指令  $OP(IR) \rightarrow CU$
- 执行指令  $Ad(IR) \rightarrow MAR \rightarrow M \rightarrow MDR \rightarrow ACC$
- $\vdots$
- 打印结果
- 停机



# 1.3 计算机硬件的主要技术指标

1. 机器字长 CPU 一次能处理数据的位数  
与 CPU 中的 寄存器位数 有关

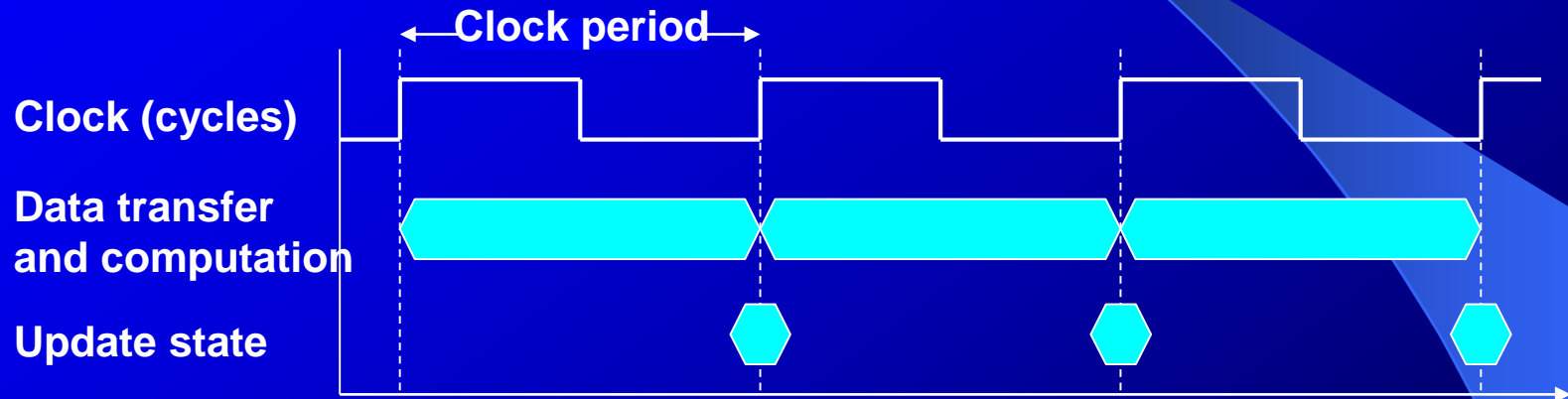
2. 运算速度 { 主频  
吉普森法  $T_M = \sum_{i=1}^n f_i t_i$  比例 执行时间  
MIPS 每秒执行百万条指令  
CPI 执行一条指令所需时钟周期数  
FLOPS 每秒浮点运算次数



# 主频

## 1.3

- **Operation of digital hardware governed by a constant-rate clock**



- **Clock period: duration of a clock cycle**
  - e.g.,  $250\text{ps} = 0.25\text{ns} = 250 \times 10^{-12}\text{s}$
- **Clock frequency (rate): cycles per second**
  - e.g.,  $4.0\text{GHz} = 4000\text{MHz} = 4.0 \times 10^9\text{Hz}$

**Clock Cycles = Instruction Count × Cycles per Instruction**

**CPU Time = Instruction Count × CPI × Clock Cycle Time**

$$= \frac{\text{Instruction Count} \times \text{CPI}}{\text{Clock Rate}}$$

- **Instruction Count for a program**
  - Determined by program, ISA and compiler
- **Average cycles per instruction**
  - Determined by CPU hardware
  - If different instructions have different CPI
    - Average CPI affected by instruction mix



# CPI例子

## 1.3

- Computer A: Cycle Time = 250ps, CPI = 2.0
- Computer B: Cycle Time = 500ps, CPI = 1.2
- Same ISA
- Which is faster, and by how much?

$$\begin{aligned}\text{CPU Time}_A &= \text{Instruction Count} \times \text{CPI}_A \times \text{Cycle Time}_A \\ &= 1 \times 2.0 \times 250\text{ps} = 1 \times 500\text{ps} \leftarrow \text{A is faster...}\end{aligned}$$

$$\begin{aligned}\text{CPU Time}_B &= \text{Instruction Count} \times \text{CPI}_B \times \text{Cycle Time}_B \\ &= 1 \times 1.2 \times 500\text{ps} = 1 \times 600\text{ps}\end{aligned}$$

$$\frac{\text{CPU Time}_B}{\text{CPU Time}_A} = \frac{1 \times 600\text{ps}}{1 \times 500\text{ps}} = 1.2 \leftarrow \text{...by this much}$$



# 深入理解CPI

## 1.3

- If different instruction classes take different numbers of cycles

$$\text{Clock Cycles} = \sum_{i=1}^n (\text{CPI}_i \times \text{Instruction Count}_i)$$

### ■ Weighted average CPI

$$\text{CPI} = \frac{\text{Clock Cycles}}{\text{Instruction Count}} = \sum_{i=1}^n \left( \text{CPI}_i \times \frac{\text{Instruction Count}_i}{\text{Instruction Count}} \right)$$

Relative frequency



# CPI例子

## 1.3

- Alternative compiled code sequences using instructions in classes A, B, C

Class	A	B	C
CPI for class	1	2	3
IC in sequence 1	2	1	2
IC in sequence 2	4	1	1

### ■ Sequence 1: IC = 5

- Clock Cycles  
 $= 2 \times 1 + 1 \times 2 + 2 \times 3$   
 $= 10$
- Avg. CPI =  $10/5 = 2.0$

### ■ Sequence 2: IC = 6

- Clock Cycles  
 $= 4 \times 1 + 1 \times 2 + 1 \times 3$   
 $= 9$
- Avg. CPI =  $9/6 = 1.5$

# MIPS与CPI

## 1.3

- **MIPS: Millions of Instructions Per Second**
  - Doesn't account for
    - Differences in ISAs between computers
    - Differences in complexity between instructions

$$\begin{aligned}\text{MIPS} &= \frac{\text{Instruction count}}{\text{Execution time} \times 10^6} \\ &= \frac{\text{Instruction count}}{\frac{\text{Instruction count} \times \text{CPI}}{\text{Clock rate}} \times 10^6} = \frac{\text{Clock rate}}{\text{CPI} \times 10^6}\end{aligned}$$

- **CPI varies between programs on a given CPU**

# MIPS与CPI

## 1.3

- 例：某计算机的主频为1.2GHz，其指令分为4类，他们在某程序中所占的比例及CPI如下表所示。该机器的MIPS数是多少？

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

- 解： $CPI=2 \times 0.5 + 3 \times 0.2 + 4 \times 0.1 + 5 \times 0.2 = 3$   
 $MIPS = 1200 / 3 = 400$

### 3. 存储容量 存放二进制信息的总位数

存储单元个数  $\times$  存储字长

如 MAR MDR 容量

10 8 1 K  $\times$  8位

16 32 64 K  $\times$  32位

$$1\text{K} = 2^{10}$$

$$2^{13} = 1\text{ KB} \quad 1\text{B} = 2^3\text{b}$$

$$2^{21} = 256\text{ KB}$$

主存容量

字节数

如

辅存容量

字节数

80 GB

$$1\text{GB} = 2^{30}\text{b}$$

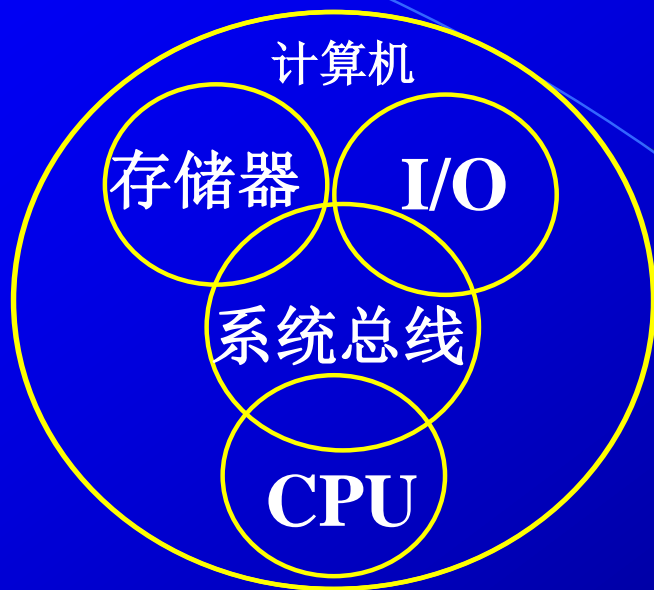
# 1.4 课程结构

计算机

第 1 篇 概论



# 1.4 课程结构



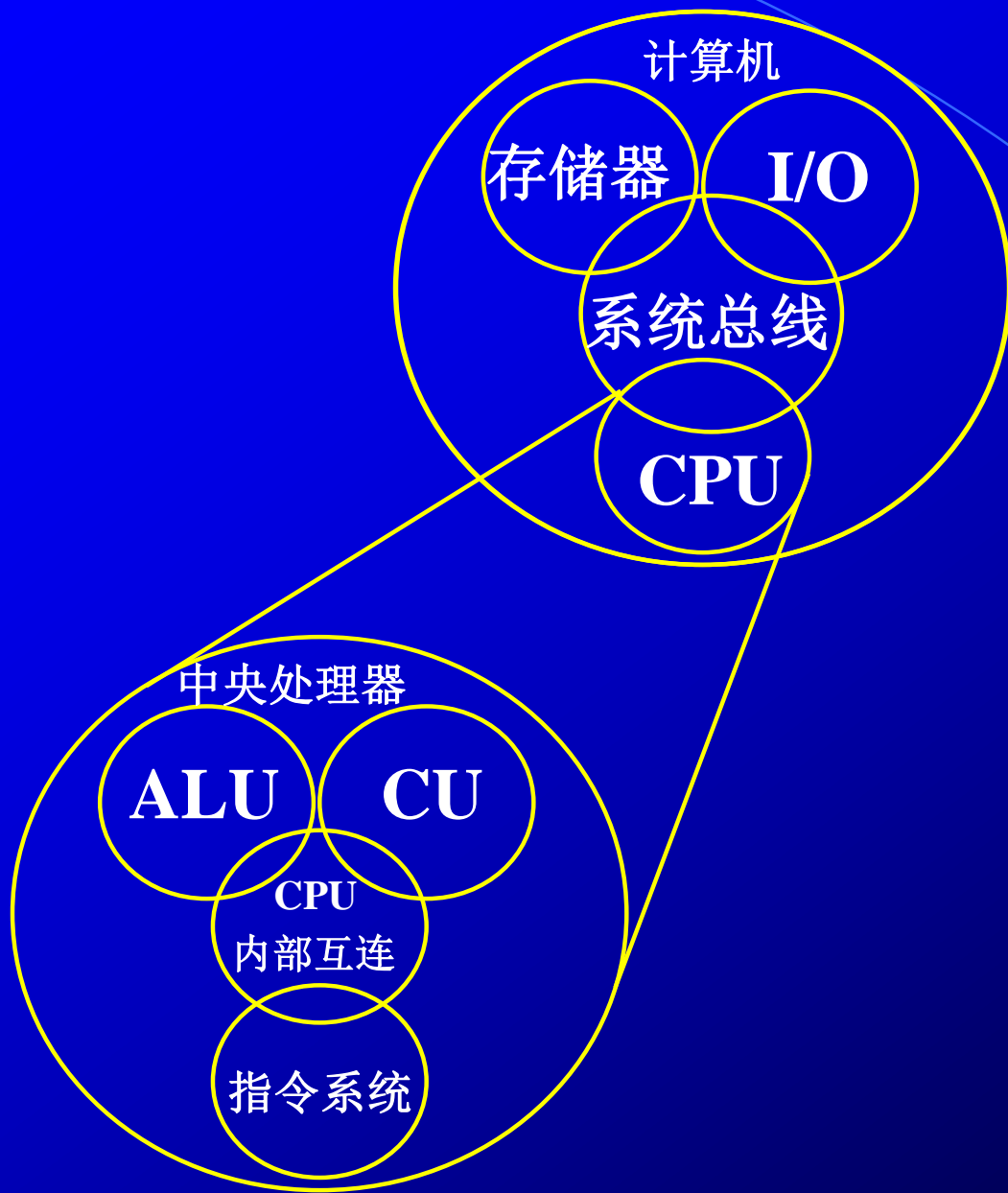
## 第 2 篇

### 计算机系统的硬件结构



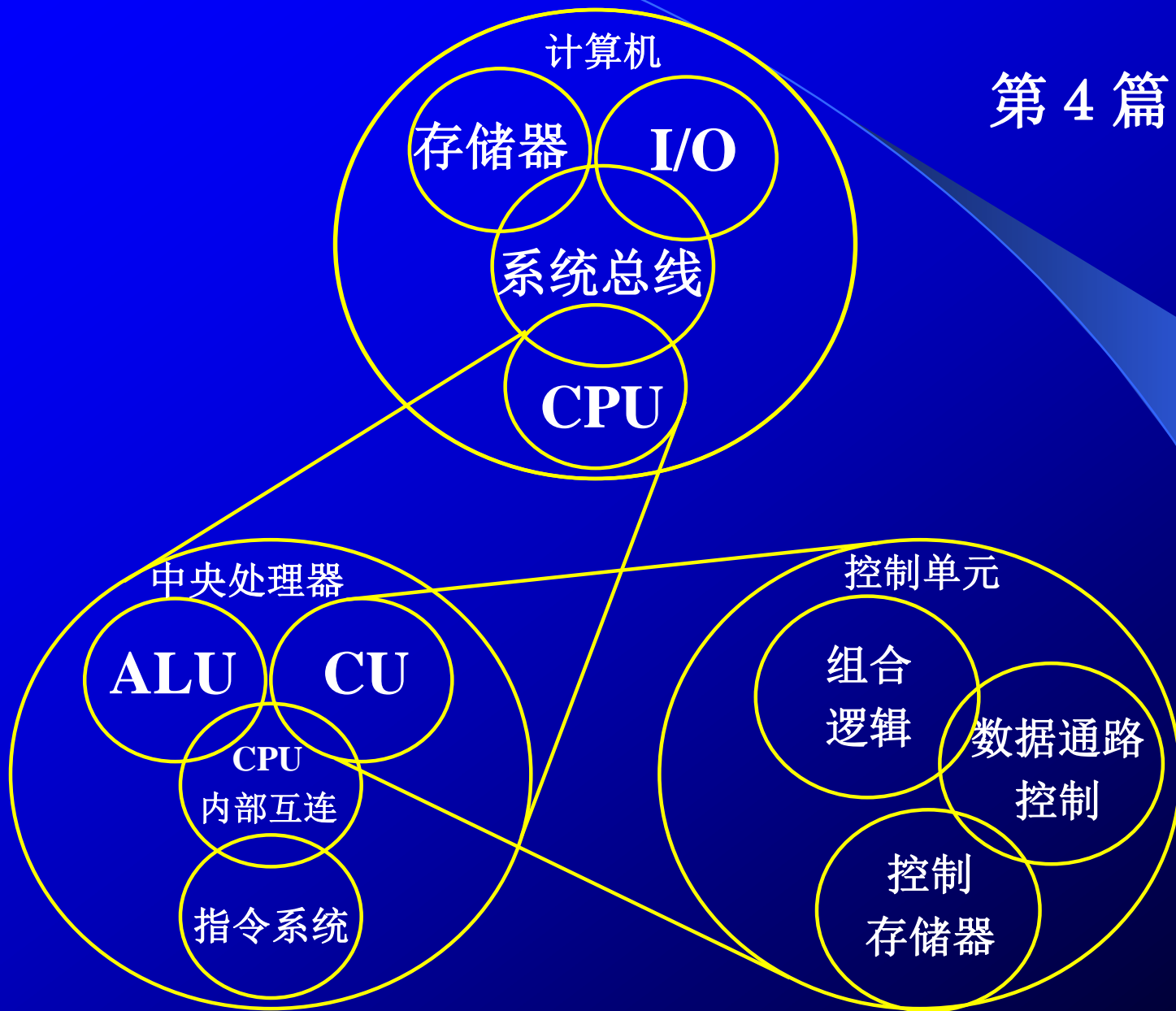
# 1.4 课程结构

## 第 3 篇 CPU



# 1.4 课程结构

## 第 4 篇 CU



# 作业

- 习题： 1.7, 1.9, 1.10

