

计算机专业导论

战德臣

哈尔滨工业大学 教授·博士生导师
教育部大学计算机课程教学指导委员会委员



Research Center on **I**ntelligent
Computing for **E**nterprises & **S**ervices,
Harbin **I**nstitute of **T**echnology

第3讲 冯.诺依曼计算机： 机器级程序及其执行

战德臣

哈尔滨工业大学 教授.博士生导师
教育部大学计算机课程教学指导委员会委员



Research Center on **I**ntelligent
Computing for **E**nterprises & **S**ervices,
Harbin **I**nstitute of **T**echnology

本讲学习什么？

---冯.诺依曼计算机：机器级程序及其执行

战德臣

哈尔滨工业大学 教授.博士生导师
教育部大学计算机课程教学指导委员会委员



Research Center on **I**ntelligent
Computing for **E**nterprises & **S**ervices,
Harbin **I**nstitute of **T**echnology

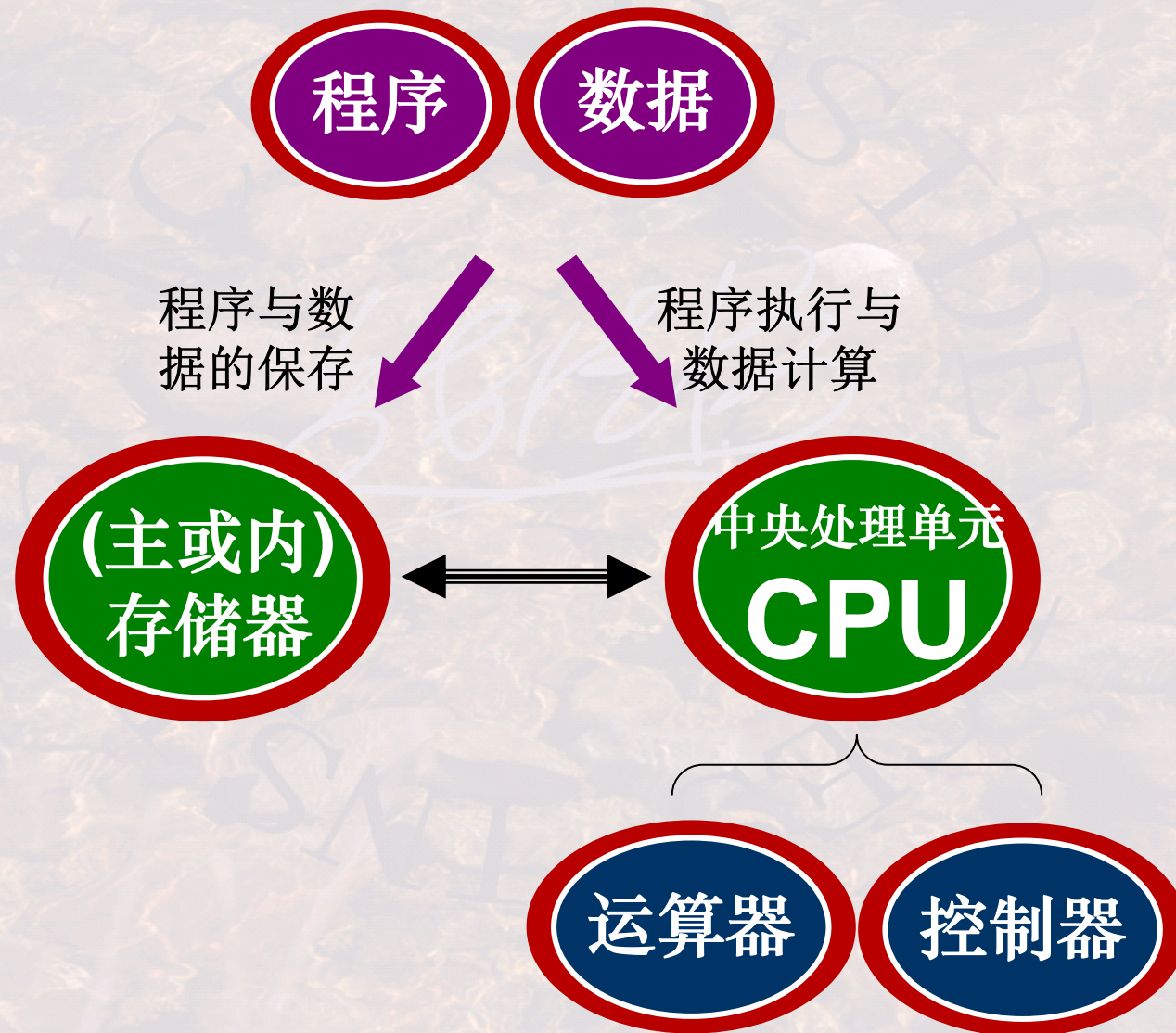
冯·诺依曼计算机：机器级程序及其执行

本讲学习的基本思路？



冯·诺依曼计算机：机器级程序及其执行

本讲学习的基本思路？



冯·诺依曼计算机：机器级程序及其执行

本讲学习的基本思路？

一台简单的
计算机

运算器

算术、逻辑
及移位运算

(数据)寄存器

R_1

R_0

00000000 00000011

00000000 00001000

00000000 00000000

00000000 00000000

控制器

时钟与节
拍发生器

信号控制

PC

00000000 00000000

IR

000001 0000001000

内容

00000100 00001000

存储单元的地址

存储单元的内容

操作码 地址码

00000000 00000000

000001

0000001000

00000000 00000001

000100

0000001001

00000000 00000010

000011

0000001010

00000000 00000011

000100

0000001000

00000000 00000100

000111

0000001011

00000000 00000101

000010

0000001100

00000000 00000110

000101

0000001100

00000000 00000111

00110

00000000 00001000

000000 0000000011

00000000 00001001

000000 0000001000

00000000 00001010

000000 0000000010

00000000 00001011

000000 00000000110

00000000 00001100

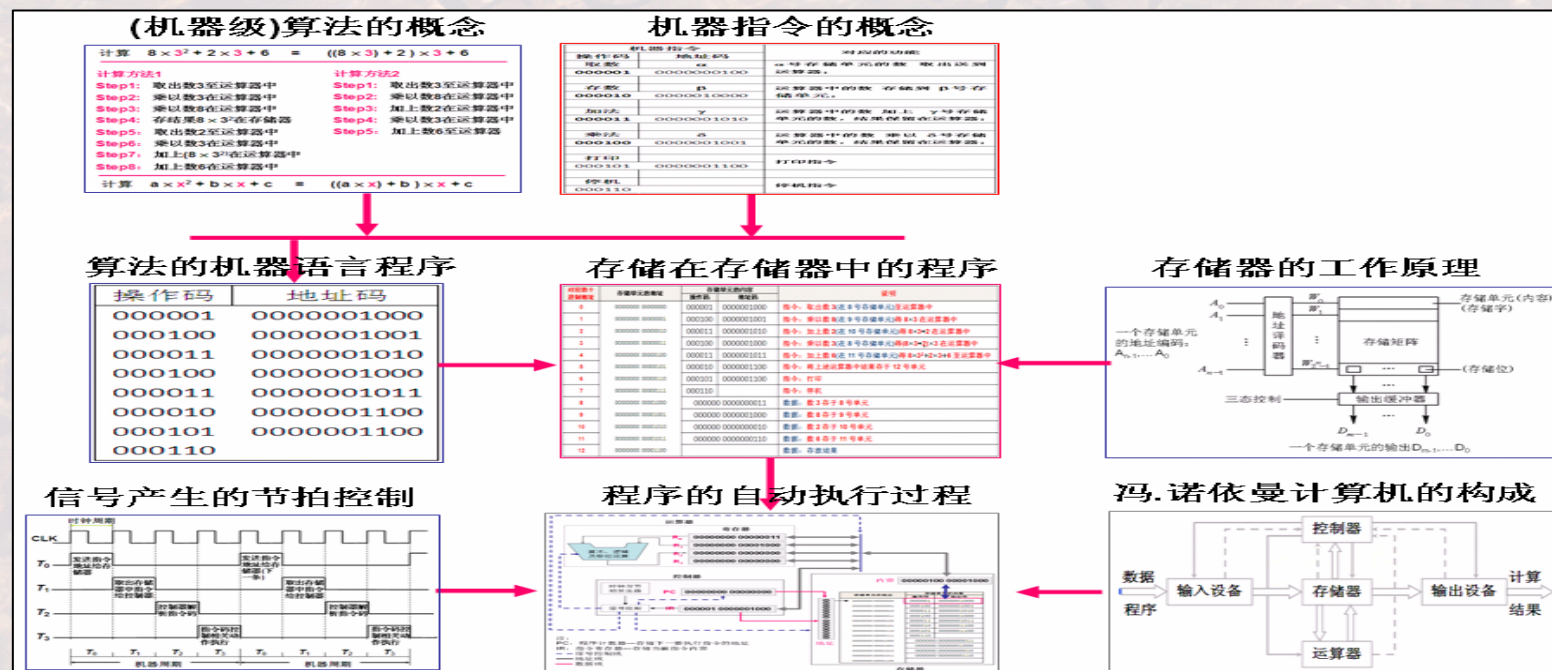
存储器
(内存)

地址

冯·诺依曼计算机：机器级程序及其执行

本讲学习的基本思路？

基本目标: 理解程序是如何被执行的



基本思维: 机器级算法与程序 → 机器指令与指令系统 → 存储器 → 存储程序 → 运算器与控制器 → 机器级程序的执行; 算法程序化 → 程序指令化 → 指令存储化 → 执行信号化

冯.诺依曼计算机：思想与构成

战德臣

哈尔滨工业大学 教授.博士生导师
教育部大学计算机课程教学指导委员会委员



Research Center on **I**ntelligent
Computing for **E**nterprises & **S**ervices,
Harbin **I**nstitute of **T**echnology

冯·诺依曼计算机: 思想与构成

(1)什么是冯·诺依曼计算机?

冯·诺依曼(Von.Neumann)计算机

◆1944~1945年间, 冯·诺伊曼提出“存储程序”的计算机设计思想, 并进行了实践, 现代计算机普遍来讲属于冯·诺伊曼机体系。

◆冯·诺伊曼机的基本思想:

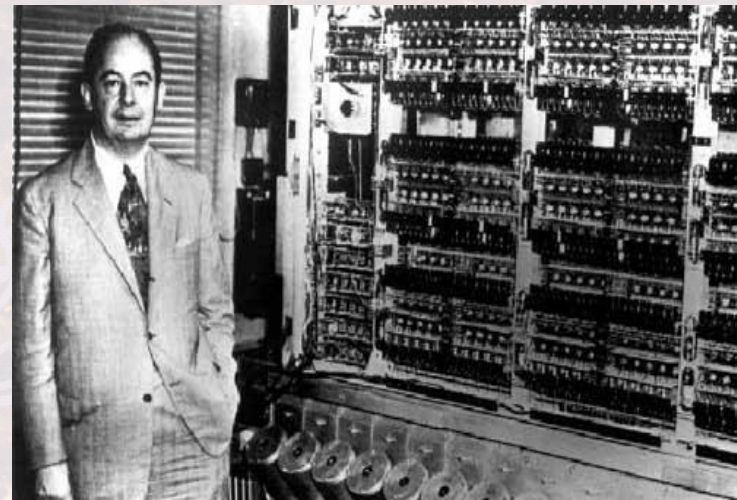
- 运算和存储分离

- 存储程序: 指令和数据以同等地位事先存于存储器, 可按地址寻访, 连续自动执行。

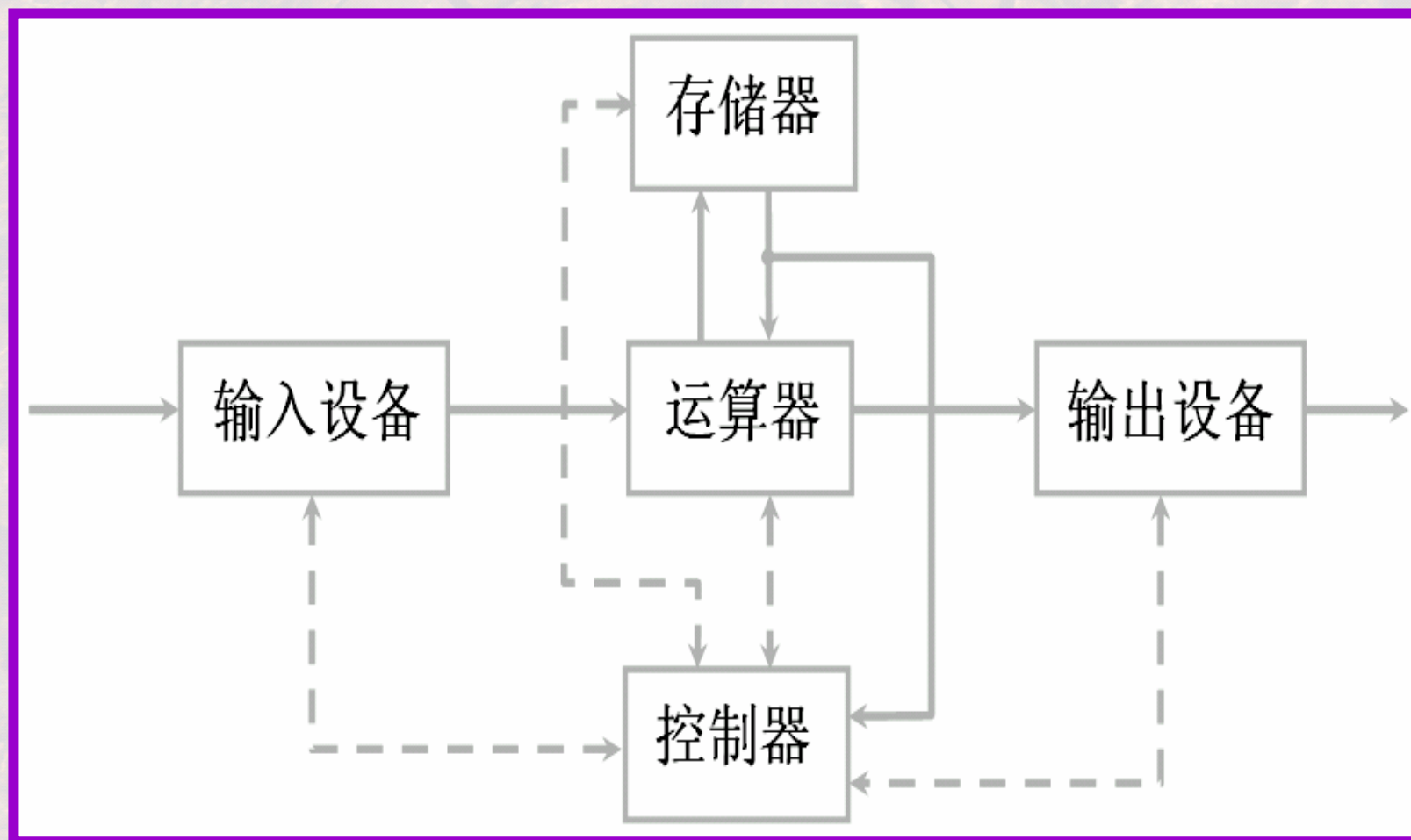
- 五大部件构成: 运算器、控制器、存储器、输入设备和输出设备

- 指令和数据用二进制表示, 指令由操作码和地址码组成

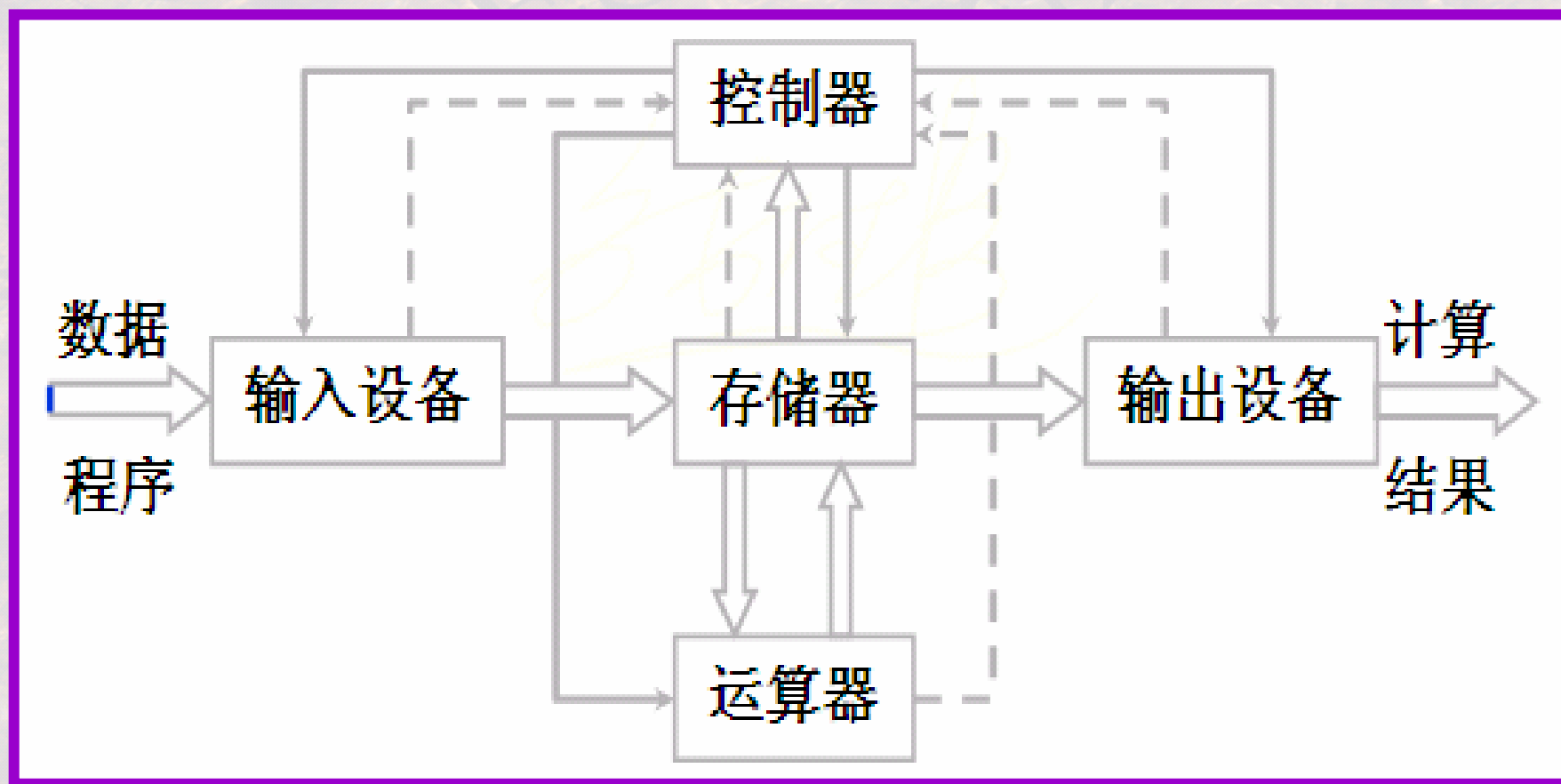
- 以运算器为中心, 控制器负责解释指令, 运算器负责执行指令



以运算器为中心的冯.诺依曼计算机构成图

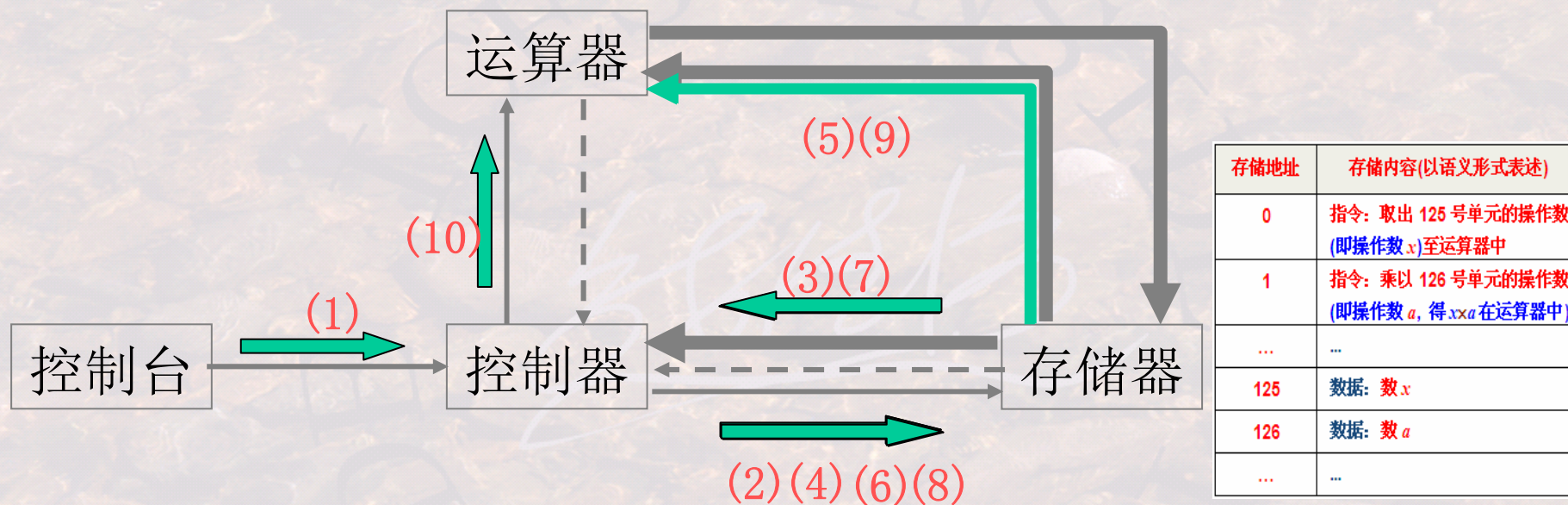


以存储器为中心的现代计算机构成图



同样是五个部件，以不同的结构来连接，便体现了不同的性能----这就是“系统”：强调“结构”，强调部件连接后的整体性、协同性

工作原理



(1) 启动控制器工作

(2) 发送第1条指令地址

(3) 取出指令并分析指令

(4) 执行指令: 发送操作数 x 所在地址

(5) 执行指令: 取出操作数 x

(6) 发送下一条指令地址

(7) 取出指令并分析指令

(8) 执行指令: 发送操作数 a 所在地址

(9) 执行指令: 取出操作数 a

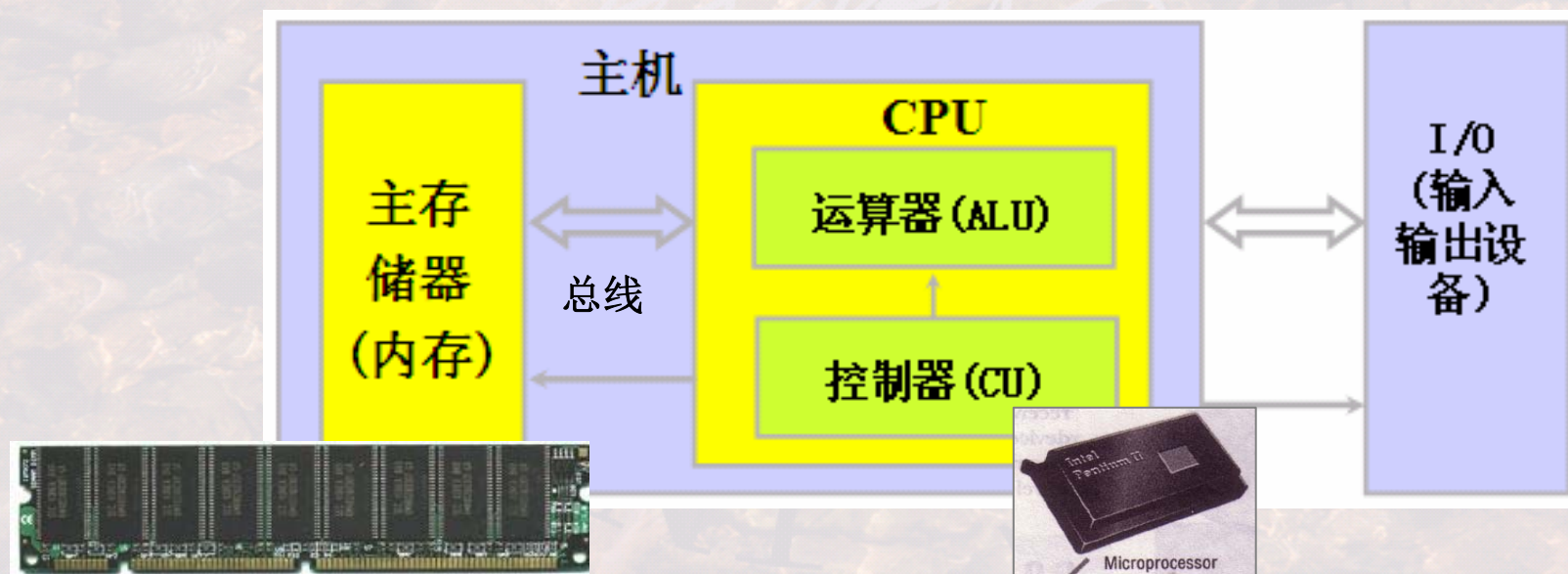
(10) 执行指令: 通知运算器计算 $a \times x$

(11) 继续后续指令的取指、执行...

计算机的基本部件

◆**CPU**: 中央处理单元(Central Process Unit), 将运算器和控制器集成在一块芯片上, 形成微处理器。

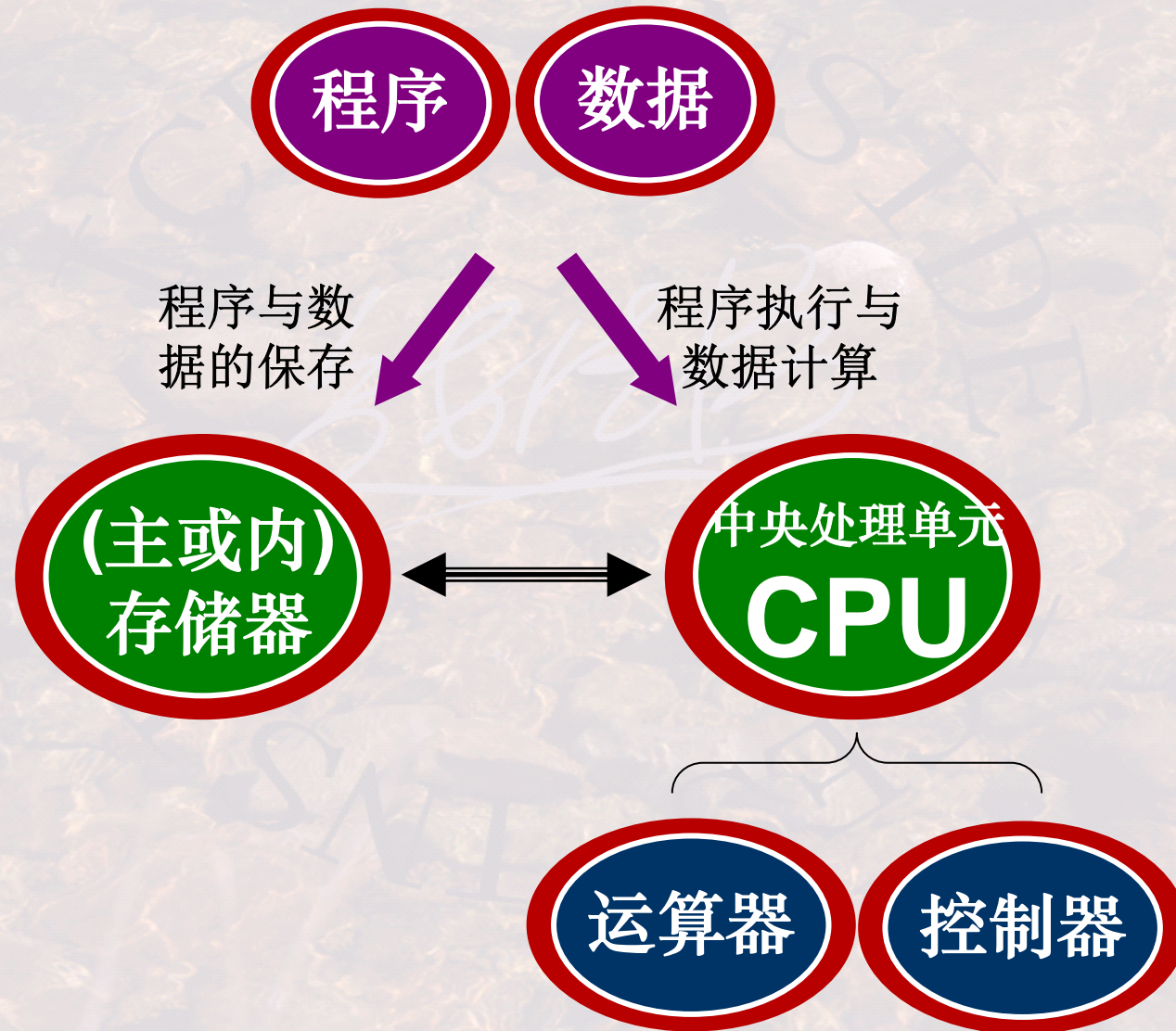
◆**CPU、主存储器、I/O设备及总线**成为现代计算机的四大核心部件。



现代计算机里面, 一个微处理器(芯片)可能包含多个**CPU**, 即多核.

冯·诺依曼计算机: 思想与构成

(6)小结



自动存取：存储器的工作原理

战德臣

哈尔滨工业大学 教授·博士生导师
教育部大学计算机课程教学指导委员会委员

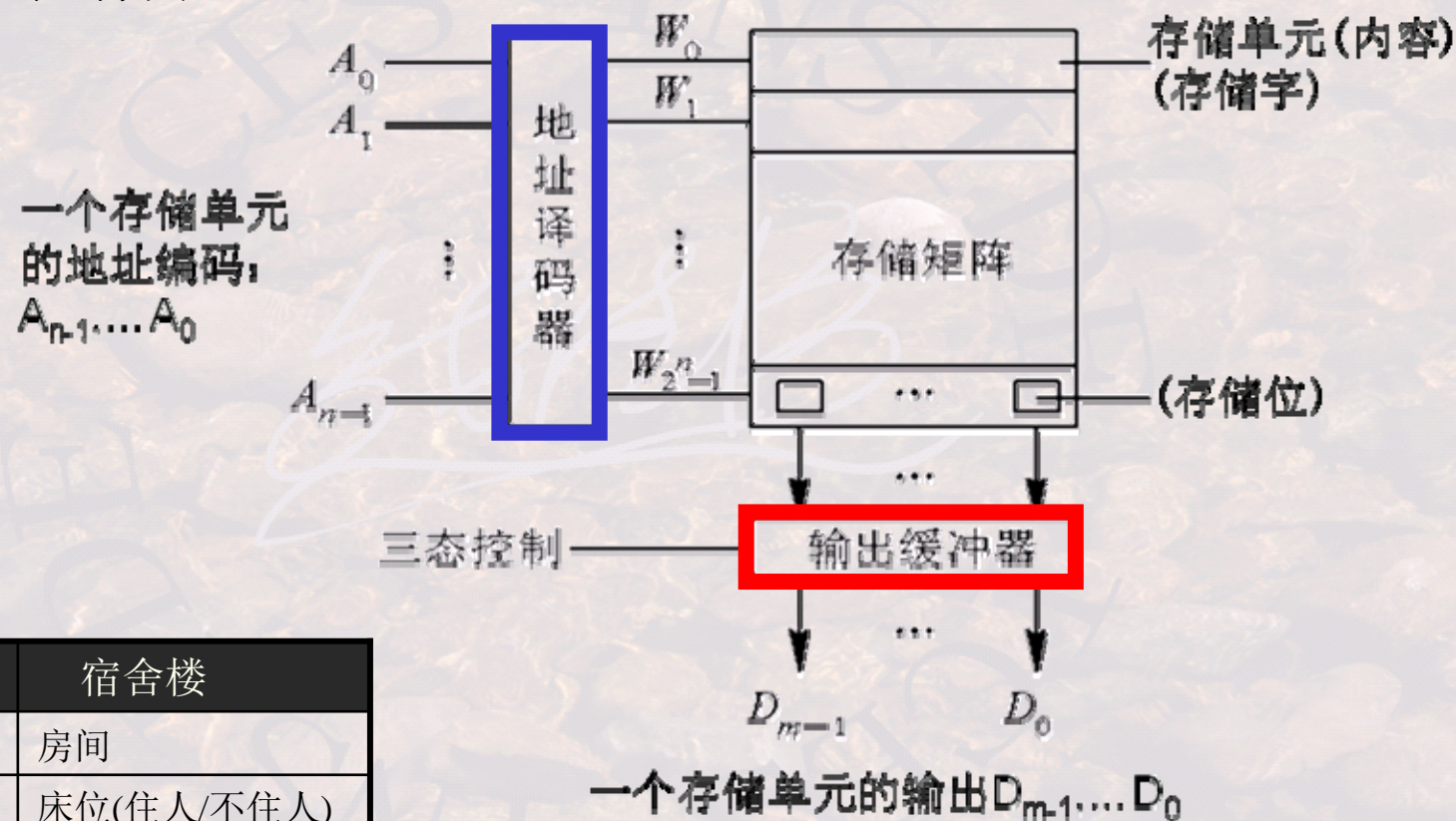


Research Center on **I**ntelligent
Computing for **E**nterprises & **S**ervices,
Harbin **I**nstitute of **T**echnology

自动存取：存储器的工作原理

(1)什么是存储器？

存储器的基本结构



概念映射

存储器	宿舍楼
存储单元	房间
存储位(存0或存1)	床位(住人/不住人)
地址编码 $A_{n-1} \dots A_0$	房间号
单元控制线 W_i	房间钥匙
输出缓冲器	公共的走廊及大门
...	...

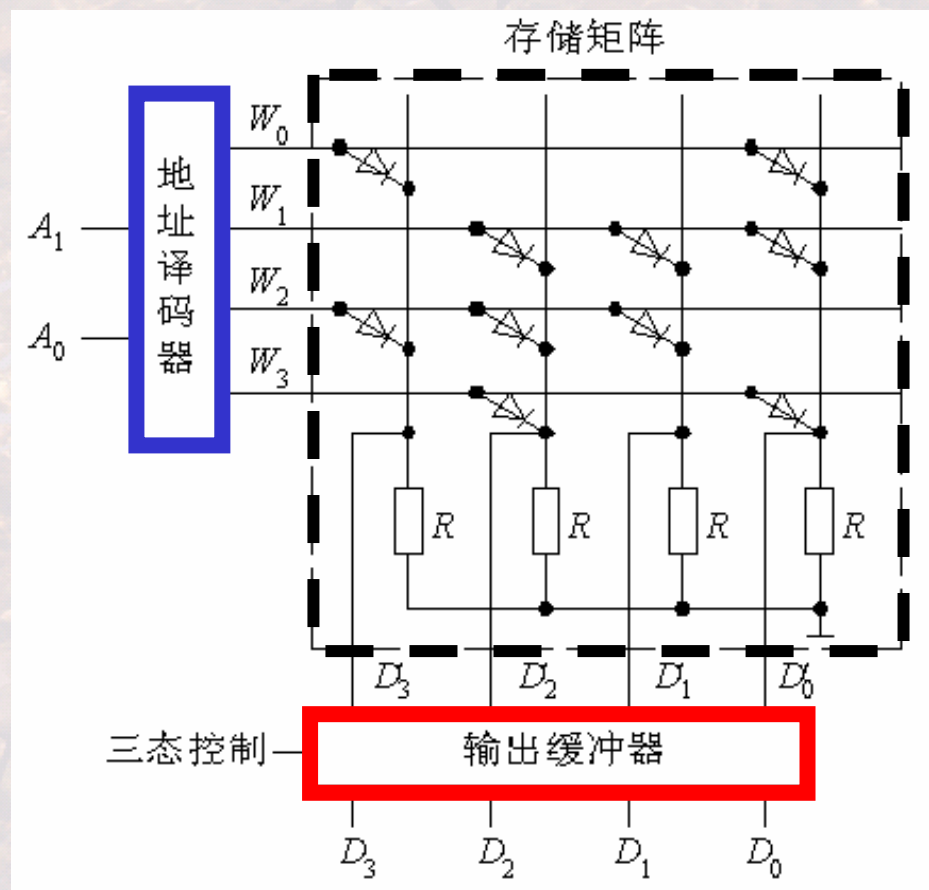
从存储器与宿舍楼的概念对比中，你能发现什么异同吗？

存储器内部的实现示例

◆当地址线和数据线间连接有二极管时，则存储的是1，否则，存储的是0

■当地址线和数据线间连接有二极管时，由地址线决定其是输出1或0，即：当地址线为高电平时，则输出1，而当地址线为低电平时，则输出0；

■没有连接的，则不受地址线影响，始终输出低电平0；

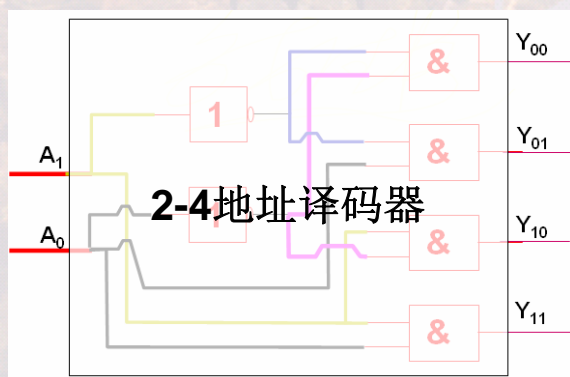


二极管ROM结构示例
(2位地址控制4个信息单元, 每个信息单元是4位0/1码)

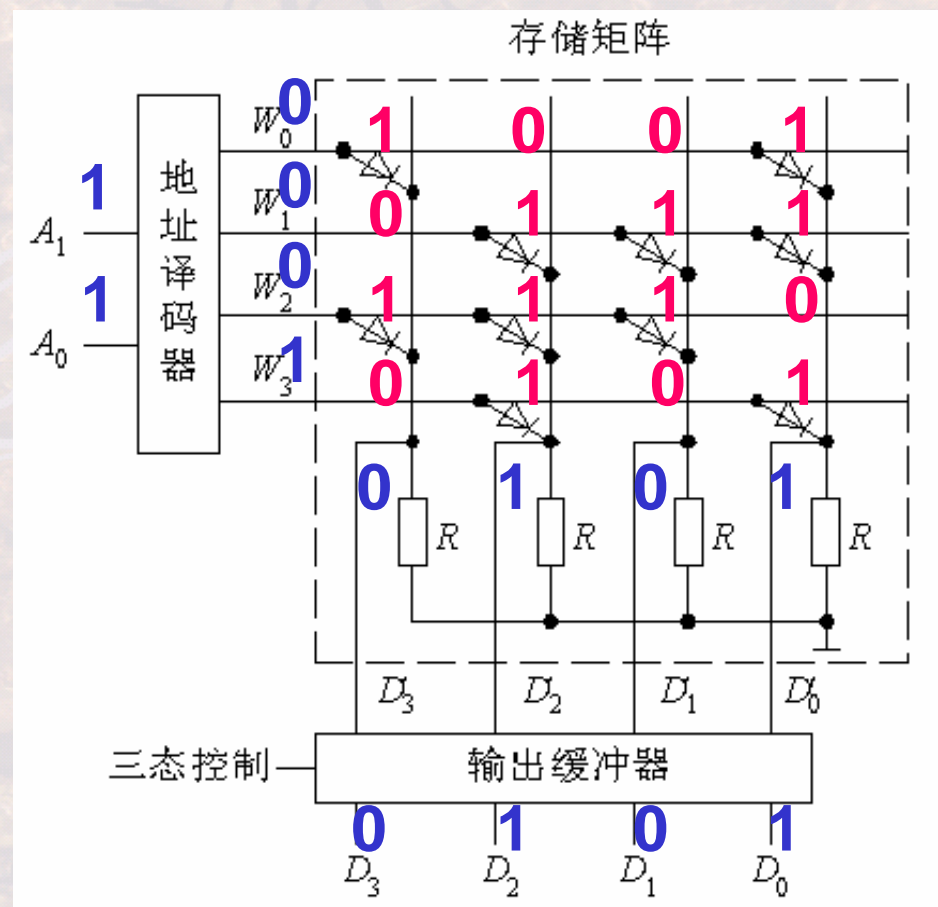
自动存取：存储器的工作原理

(2)存储器是怎样存储0和1的？又是怎样控制存取的？

存储器内部的实现示例



将地址编码转换为地址单元控制信号
类比:将房间号转换成房间钥匙



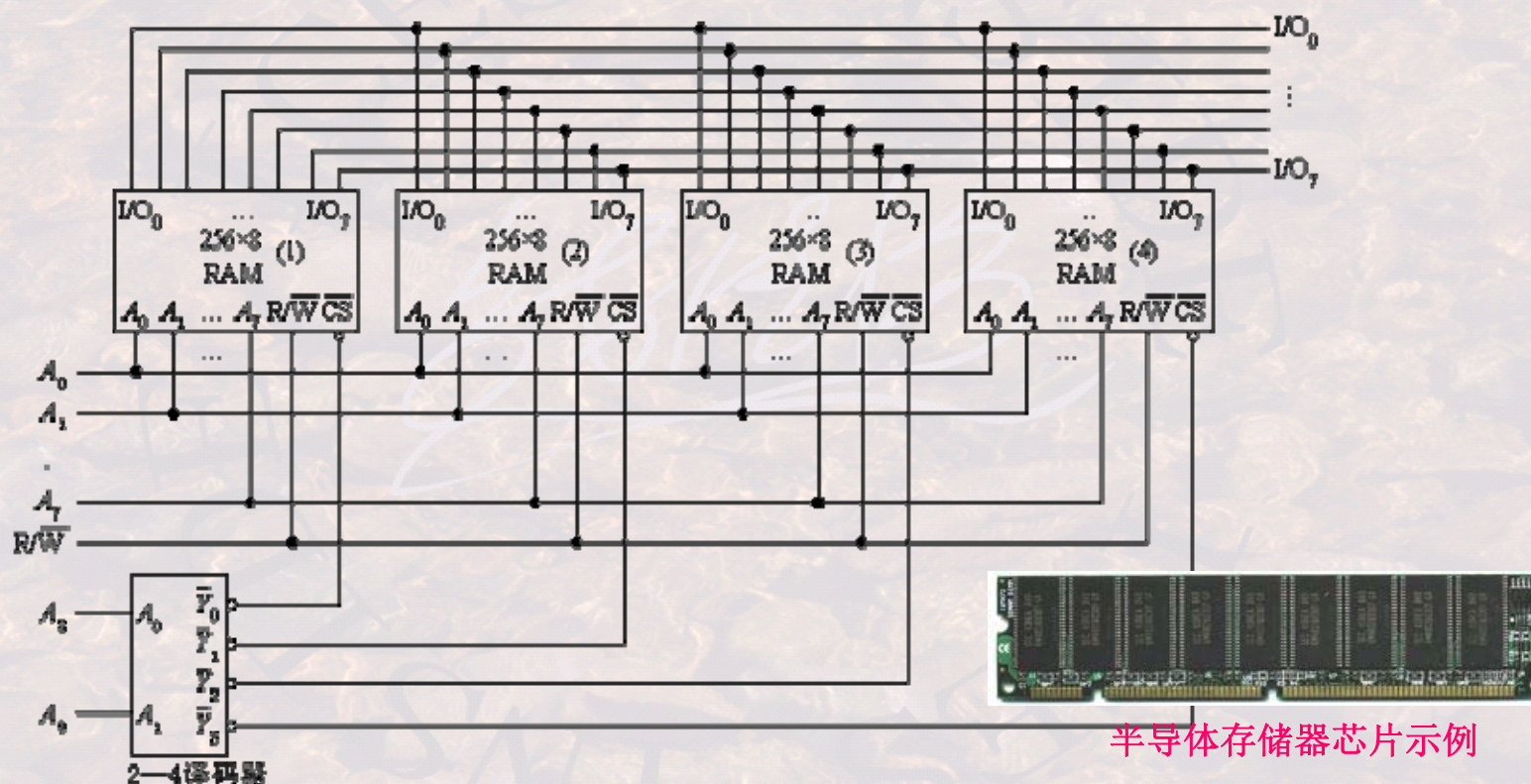
二极管ROM结构示例

(2位地址控制4个信息单元, 每个信息单元是4位0/1码)

自动存取：存储器的工作原理

(3)存储器芯片容量不够了怎么办？

用多个存储器芯片可搭建容量更大的存储器

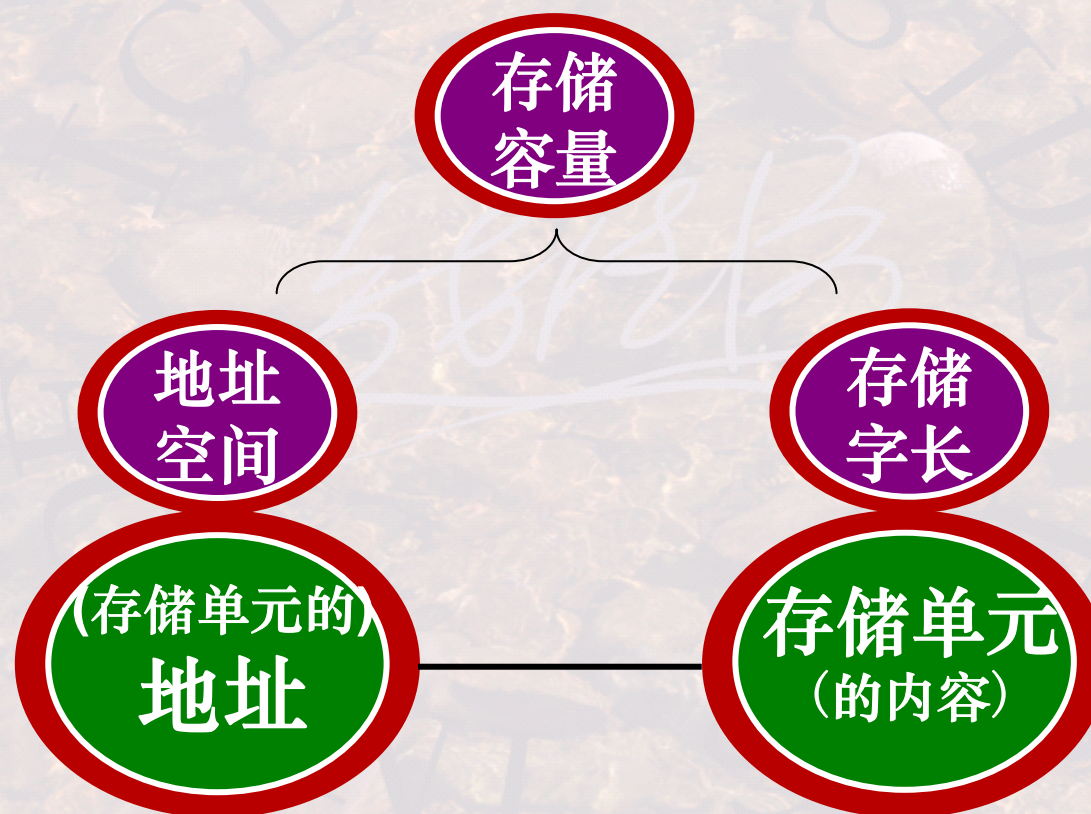


利用4个256x8存储器芯片扩展出1024x8存储器的电路图

问：从概念的角度，你能说说存储器扩展要解决什么问题吗？
提示：地址编码空间, 存储字长.

自动存取：存储器的工作原理

(4)小结？



机器指令与机器级程序

战德臣

哈尔滨工业大学 教授·博士生导师
教育部大学计算机课程教学指导委员会委员



Research Center on **I**ntelligent
Computing for **E**nterprises & **S**ervices,
Harbin **I**nstitute of **T**echnology

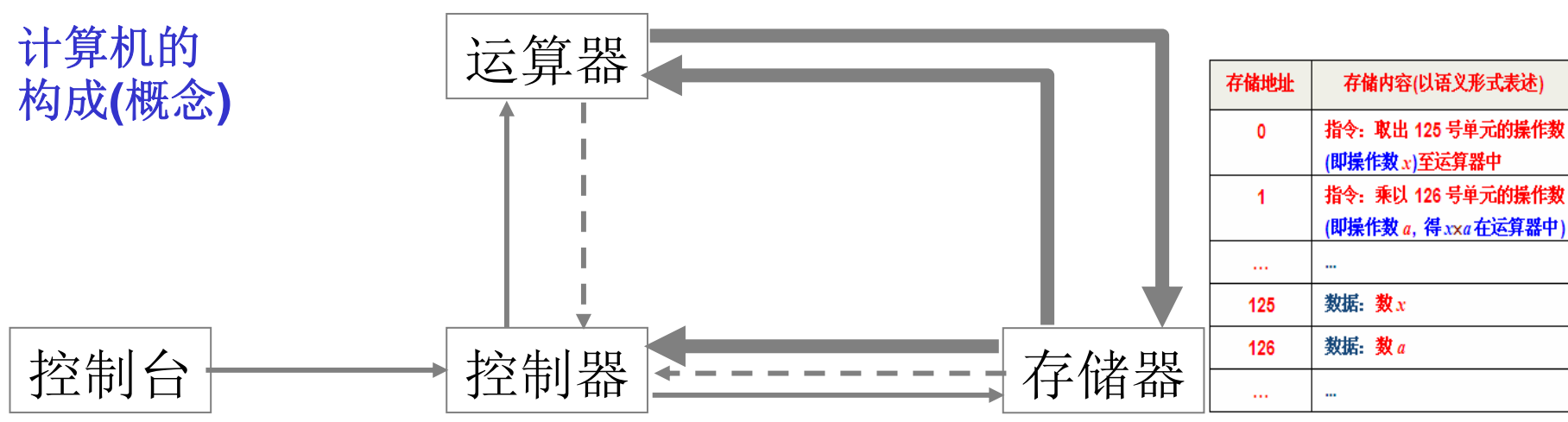
机器指令与机器级程序

(1)如何计算一个运算式?

问题---计算机如何计算一个运算式?

$$8 \times 3^2 + 2 \times 3 + 6$$

计算机的
构成(概念)



机器指令与机器级程序

(2)什么是算法?



算法---从冯.诺依曼计算机的角度

可在机器上执行的求解问题的操作规则及步骤, 被称为可执行的算法。

$$\text{计算 } 8 \times 3^2 + 2 \times 3 + 6 = ((8 \times 3) + 2) \times 3 + 6$$

计算方法1

- Step1:** 取出数3至运算器中
- Step2:** 乘以数3在运算器中
- Step3:** 乘以数8在运算器中
- Step4:** 存结果 8×3^2 在存储器中
- Step5:** 取出数2至运算器中
- Step6:** 乘以数3在运算器中
- Step7:** 加上 (8×3^2) 在运算器中
- Step8:** 加上数6在运算器中

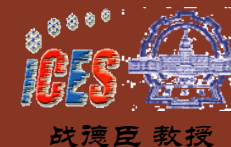
计算方法2

- Step1:** 取出数3至运算器中
- Step2:** 乘以数8在运算器中
- Step3:** 加上数2在运算器中
- Step4:** 乘以数3在运算器中
- Step5:** 加上数6至运算器中

问: 怎么看待算法节省的步数? ---算法需要“优化”

机器指令与机器级程序

(3)机器指令是怎样的？



机器指令 - 机器语言

◆ 机器指令是CPU可以直接分析并执行的指令，一般由0和1的编码表示。

◆ 指令 \approx $\underbrace{\text{操作码}}_{\text{操作码}} + \underbrace{\text{地址码}}_{\text{地址码}}$;

000001 00 00000111

000100 00 00001010

(如取数，加法等操作) (操作中的数据的来源)

000001 0000000100

000001 0000001100

000001 0000001000

机器指令		对应的功能
操作码	地址码	
取数	α	α 号存储单元的数 取出送到运算器;
000001	0000000100	
存数	β	运算器中的数 存储到 β 号存储单元;
000010	0000010000	
加法	γ	运算器中的数 加上 γ 号存储单元的数，结果保留在运算器;
000011	0000001010	
乘法	δ	运算器中的数 乘以 δ 号存储单元的数，结果保留在运算器;
000100	0000001001	
打印		打印指令
000101	0000001100	
停机		停机指令
000110		

机器指令与机器级程序

(4)怎样用机器指令表达算法?

$$((8 \times 3) + 2) \times 3 + 6$$

机器级 算法

计算方法2

- Step1:** 取出数3至运算器中
Step2: 乘以数8在运算器中
Step3: 加上数2在运算器中
Step4: 乘以数3在运算器中
Step5: 加上数6至运算器中

机器 级程序

```
000001 0000001000
000100 0000001001
000011 0000001010
000100 0000001000
000011 0000001011
000010 0000001100
000101 0000001100
000110
```

机器指令		对应的功能
操作码	地址码	
取数	α	α 号存储单元的数 取出送到运算器;
000001	0000000100	
存数	β	运算器中的数 存储到 β 号存储单元;
000010	0000010000	
加法	γ	运算器中的数 加上 γ 号存储单元的数, 结果保留在运算器;
000011	0000001010	
乘法	δ	运算器中的数 乘以 δ 号存储单元的数, 结果保留在运算器;
000100	0000001001	
打印		打印指令
000101	0000001100	
停机		停机指令
000110		

机器 指令

“3” 存储在8号存储单元
 “8” 存储在9号存储单元
 “2” 存储在10号存储单元
 “6” 存储在11号存储单元

(5)将机器级程序和数据装载进存储器中？

存储器

计算 $8 \times 3^2 + 2 \times 3 + 6$ 的程序；
计算 $ax^2 + bx + c$ 的程序。

机器级程序

对应的十进制地址	存储单元的地址	存储单元的内容		说明
		操作码	地址码	
0	00000000 00000000	000001	0000001000	指令：取出 8 号存储单元的数(即 3)至运算器中
1	00000000 00000001	000100	0000001001	指令：乘以 9 号存储单元的数(即 8)得 8×3 在运算器中
2	00000000 00000010	000011	0000001010	指令：加上 10 号存储单元的数(即 2)得 $8 \times 3 + 2$ 在运算器中
3	00000000 00000011	000100	0000001000	指令：乘以 8 号存储单元的数(即 3) 得 $(8 \times 3 + 2) \times 3$ 在运算器中
4	00000000 00000100	000011	0000001011	指令：加上 11 号存储单元的数(即 6)得 $8 \times 3^2 + 2 \times 3 + 6$ 至运算器中
5	00000000 00000101	000010	0000001100	指令：将上述运算器中结果存于 12 号存储单元
6	00000000 00000110	000101	0000001100	指令：打印
7	00000000 00000111	000110		指令：停机
8	00000000 00001000	000000	0000000011	数据：数 3 存于 8 号单元
9	00000000 00001001	000000	00000001000	数据：数 8 存于 9 号单元
10	00000000 00001010	000000	00000000010	数据：数 2 存于 10 号单元
11	00000000 00001011	000000	000000000110	数据：数 6 存于 11 号单元
12	00000000 00001100			数据：存放结果

程序与数据以同等地位存于存储器中

高级语言程序的示例

计算 ax^2+bx+c

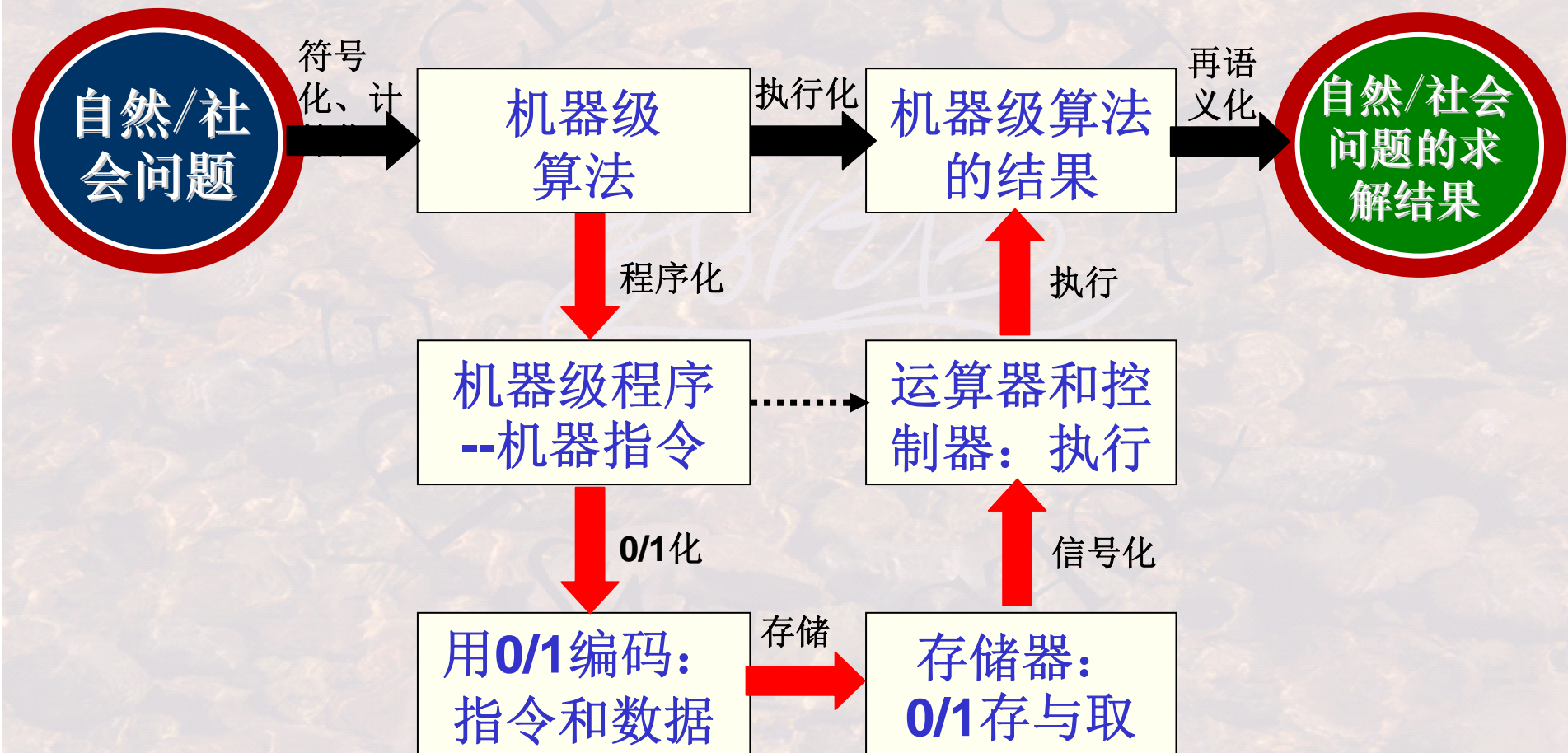
其中 a, x, b, c 是变量。

变量的地址是由编译程序在编译过程中自动分配的，也即是说编译器根据当时编译的情况，分配 a, x, b, c 为8号，9号，10号，11号存储单元，并产生上述的机器指令程序

```
Main() {  
    int result;    //定义变量 result  
    int x;         //定义变量 x  
    int a;         //定义变量 a  
    int b;         //定义变量 b  
    int c;         //定义变量 c  
    x = 3;         //将 3 赋值给 x  
    //数据赋值过程中也可在运行过程中进行  
    a = 8;         //将 8 赋值给 a  
    b = 2;         //将 2 赋值给 b  
    c = 6;         //将 6 赋值给 c  
    result = a * x * x + b * x + c;  
    //计算  $a * x * x + b * x + c$  并赋值给 result  
    print result;  //打印 result 的值  
}
```


机器指令与机器级程序

(7)小结?



机器级程序的执行机制

战德臣

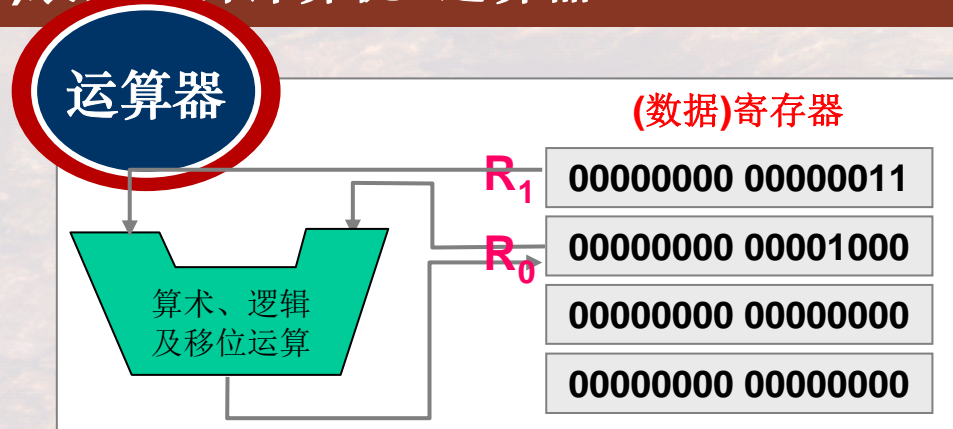
哈尔滨工业大学 教授·博士生导师
教育部大学计算机课程教学指导委员会委员



Research Center on **I**ntelligent
Computing for **E**nterprises & **S**ervices,
Harbin **I**nstitute of **T**echnology

机器级程序的执行机制

(1) 装配一台计算机--运算器



□ (数据)寄存器

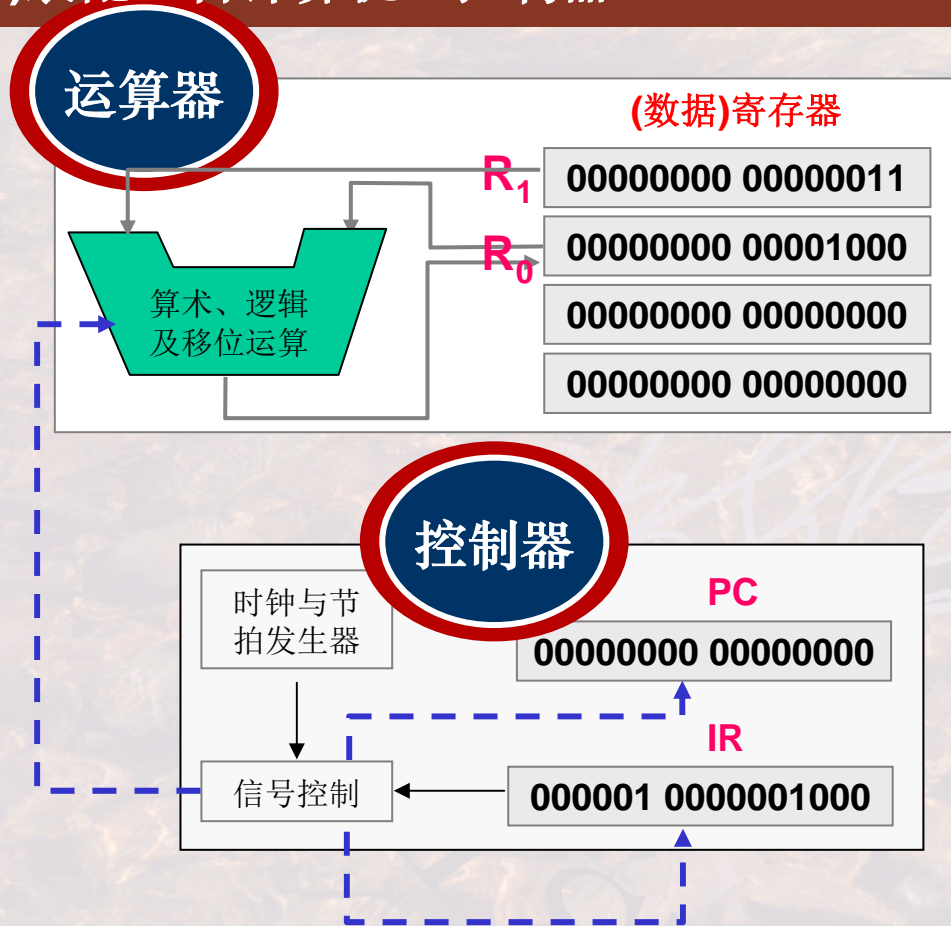
□ 算术逻辑部件

$$R_0 = R_1 \theta R_0$$

(赋值, R_0 既是一个操作数, 又保存运算结果)。
其中 θ 为算术、逻辑及移位运算符

机器级程序的执行机制

(2) 装配一台计算机—控制器



- 程序计数器PC
- 指令寄存器
- 信号控制器
- 时钟与信号发生器

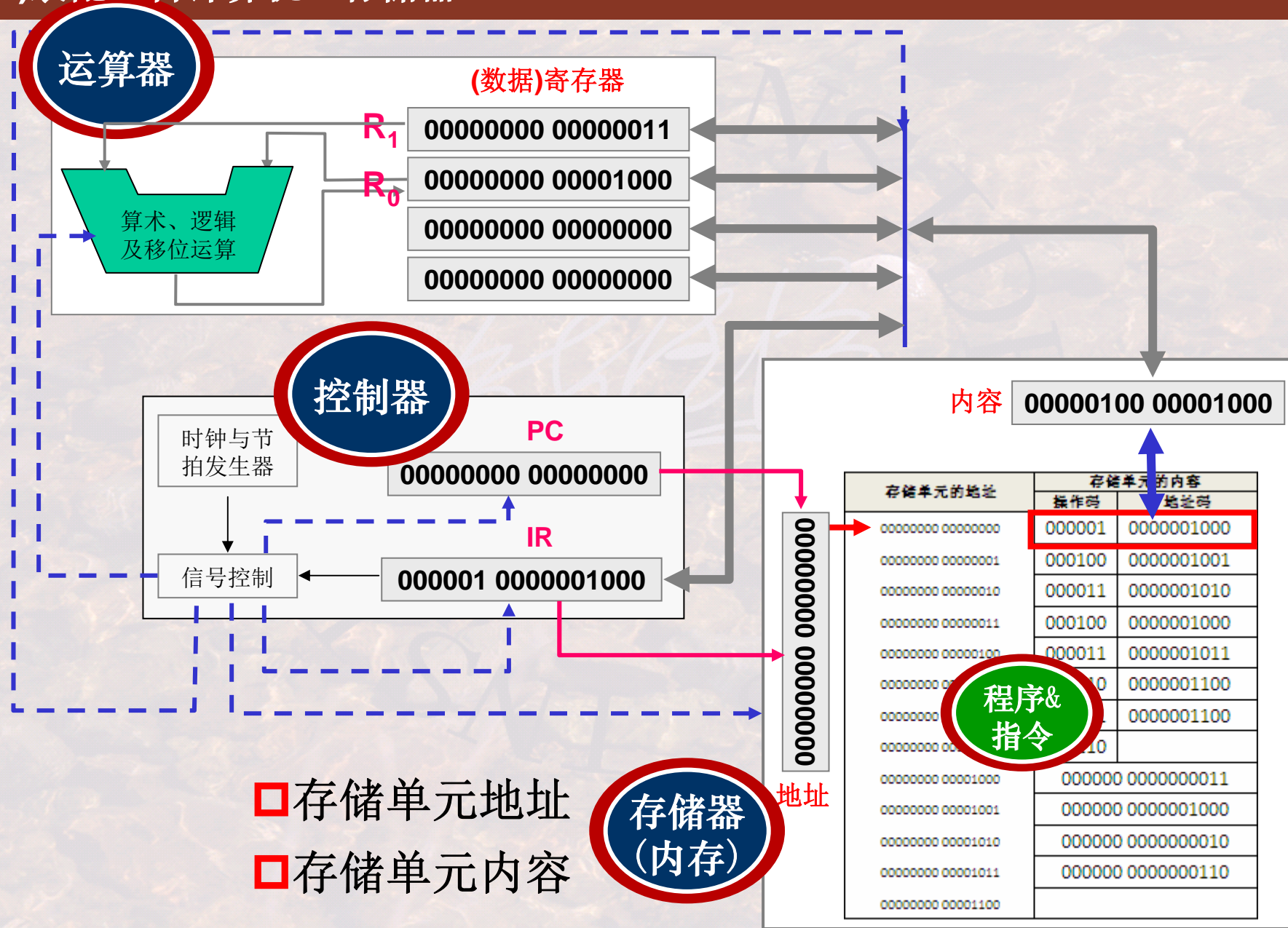
注:

PC: 程序计数器---存储下一要执行指令的地址

IR: 指令寄存器---存储当前指令内容

- 信号控制线
- 数据线
- 地址线

(3) 装配一台计算机—存储器



机器级程序的执行机制

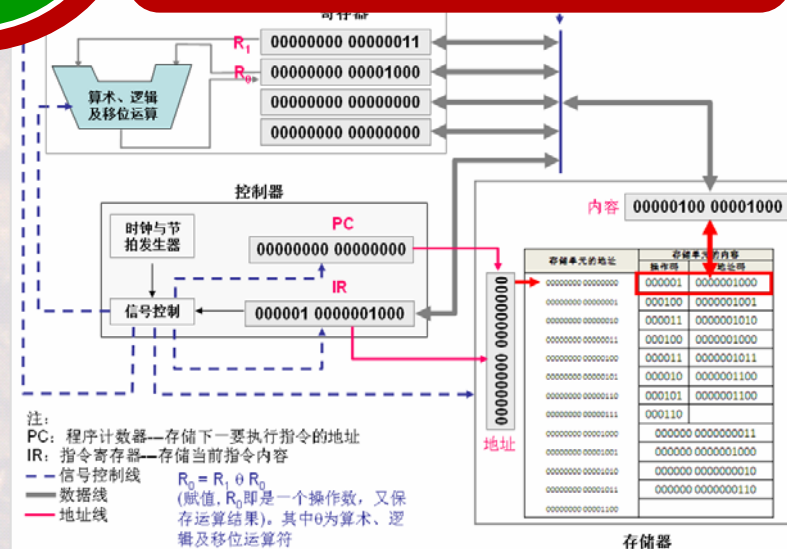
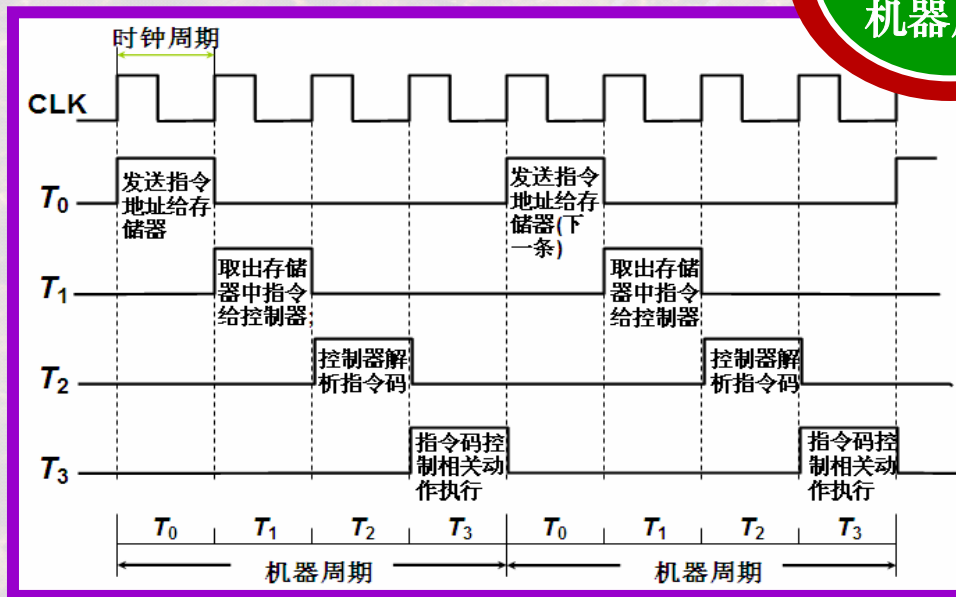
(4)指令是怎样被执行的？

指令执行

- ◆不同的指令，由一组不同的电信号构成
- ◆同一指令的电信号在时钟与节拍的控制下按次序产生与传输
- ◆一条指令占用一个或多个机器周期，一个机器周期又分为多个节拍
- ◆最小的时间区隔单位--时钟周期

时钟周期、
节拍与
机器周期

指令执行的信号化——即在节拍控制下有序地发出各种电信号



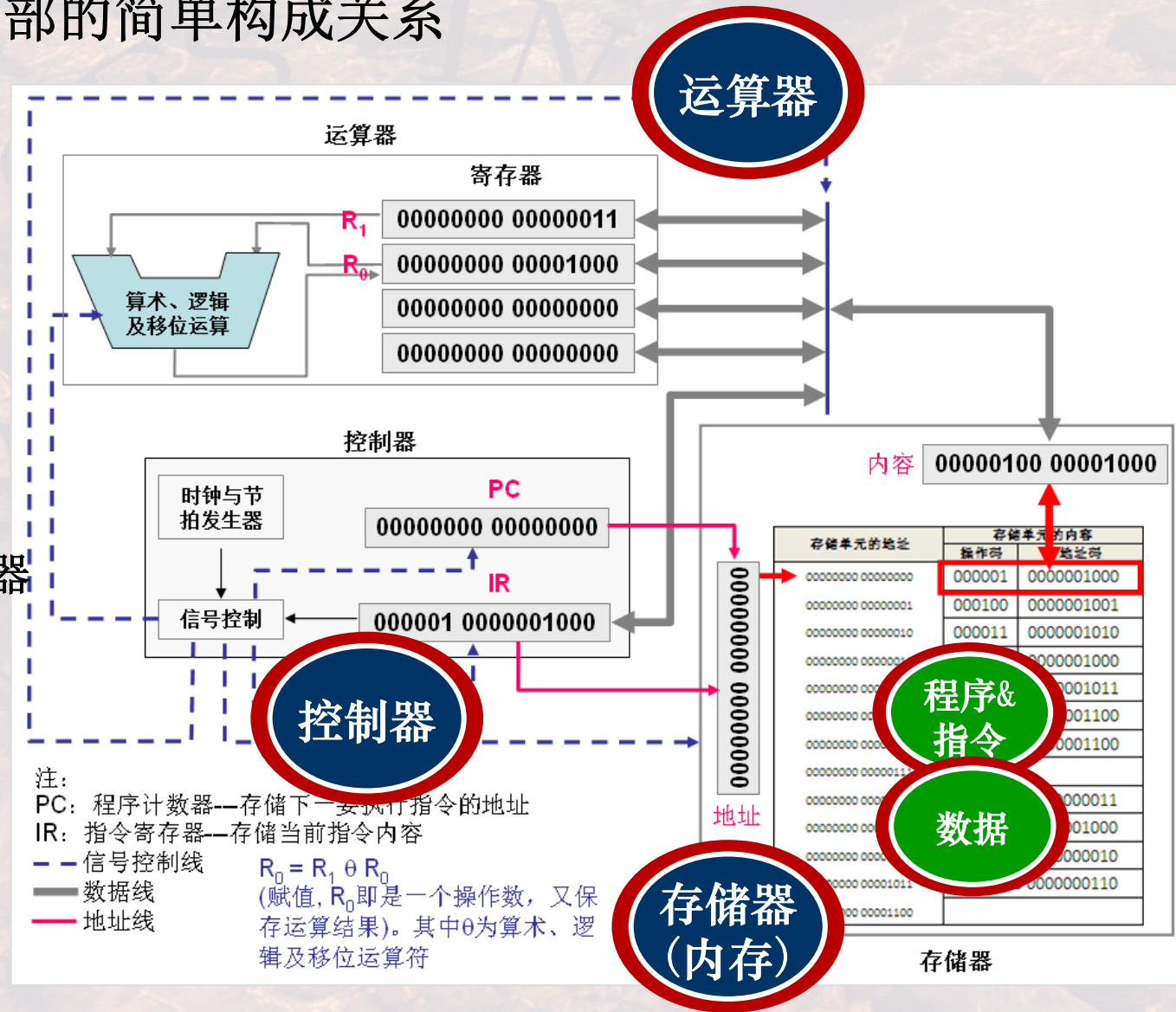
问：机器的“主频”指的是什么？

机器级程序的执行机制

(5)机器级程序被装载进存储器中

计算机各部件内部的简单构成关系

- 寄存器
- 算术逻辑部件
- 程序计数器PC
- 指令寄存器
- 信号控制器
- 时钟与信号发生器
- 存储单元地址
- 存储单元内容



机器级程序的执行过程模拟

战德臣

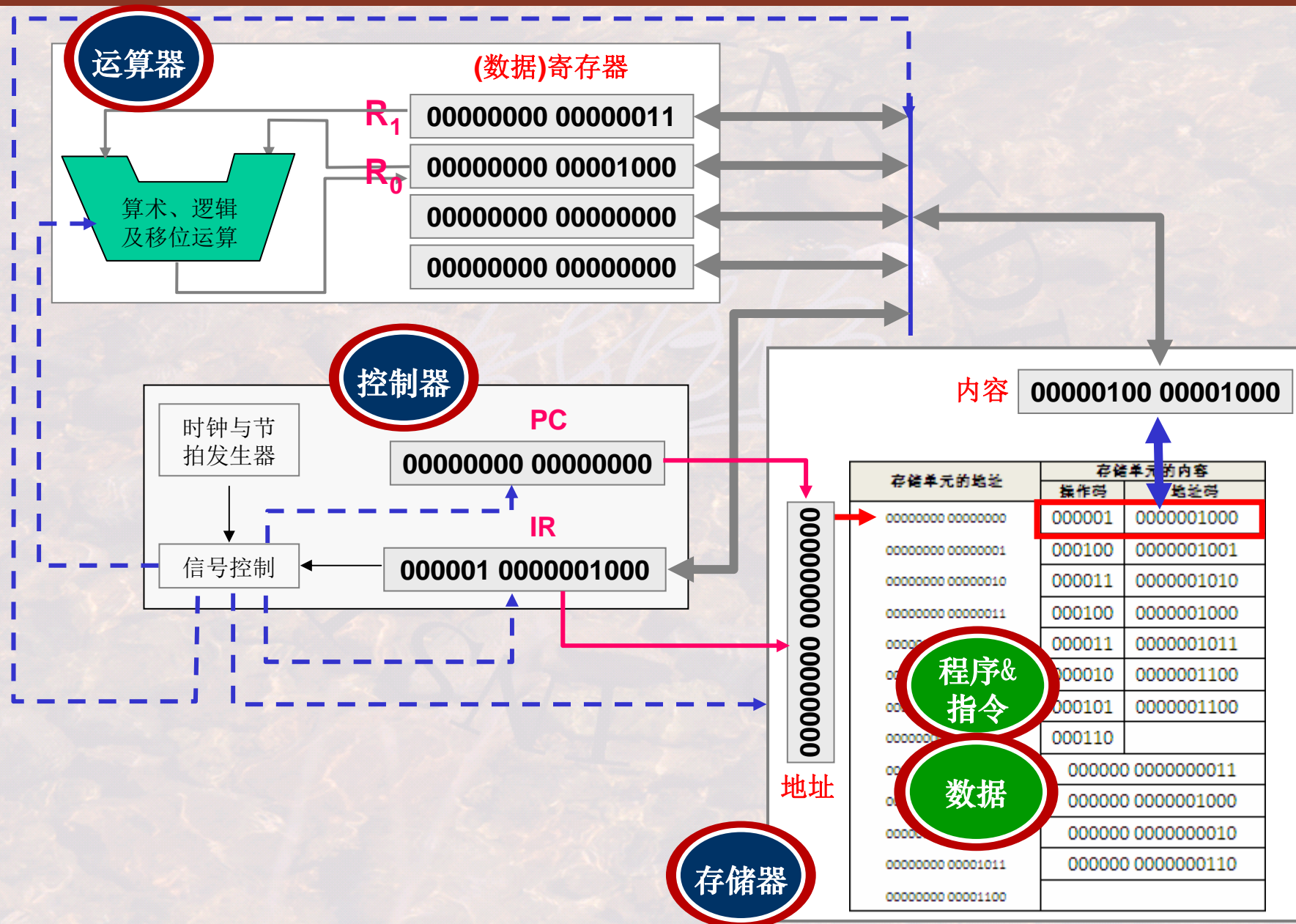
哈尔滨工业大学 教授·博士生导师
教育部大学计算机课程教学指导委员会委员



Research Center on **I**ntelligent
Computing for **E**nterprises & **S**ervices,
Harbin **I**nstitute of **T**echnology

机器级程序的执行过程模拟

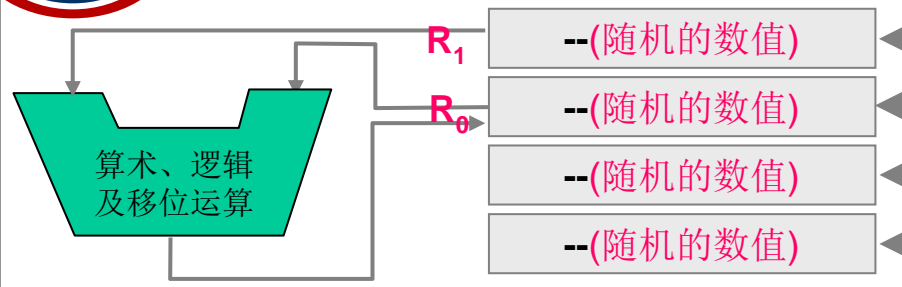
(1) 程序和数据已经装入存储器中如何执行呢？



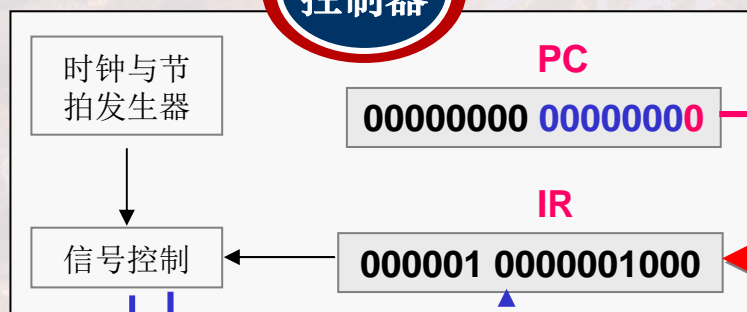
机器级程序的执行过程模拟

(2)第1条指令的读取

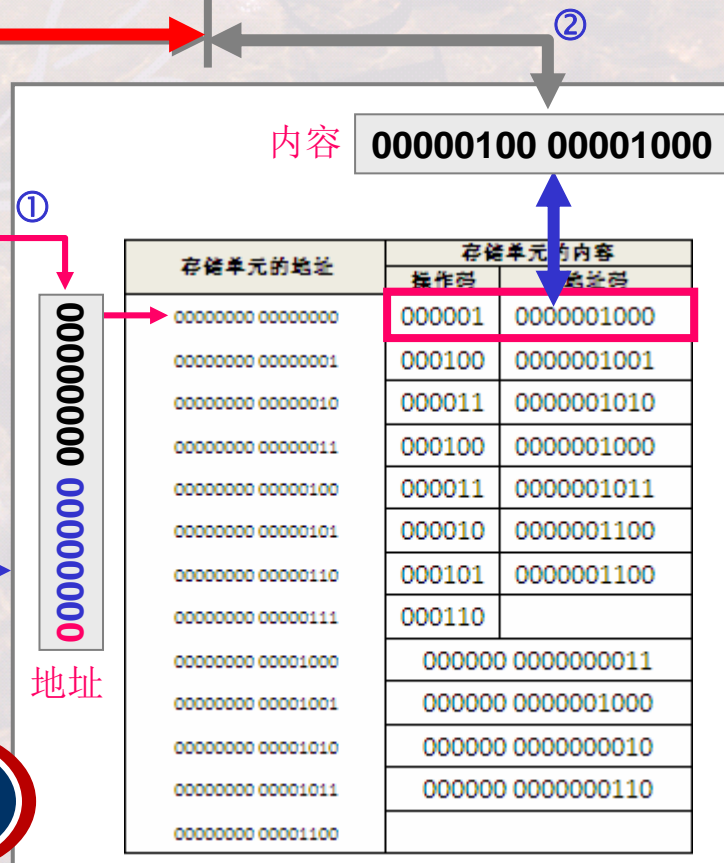
运算器



控制器

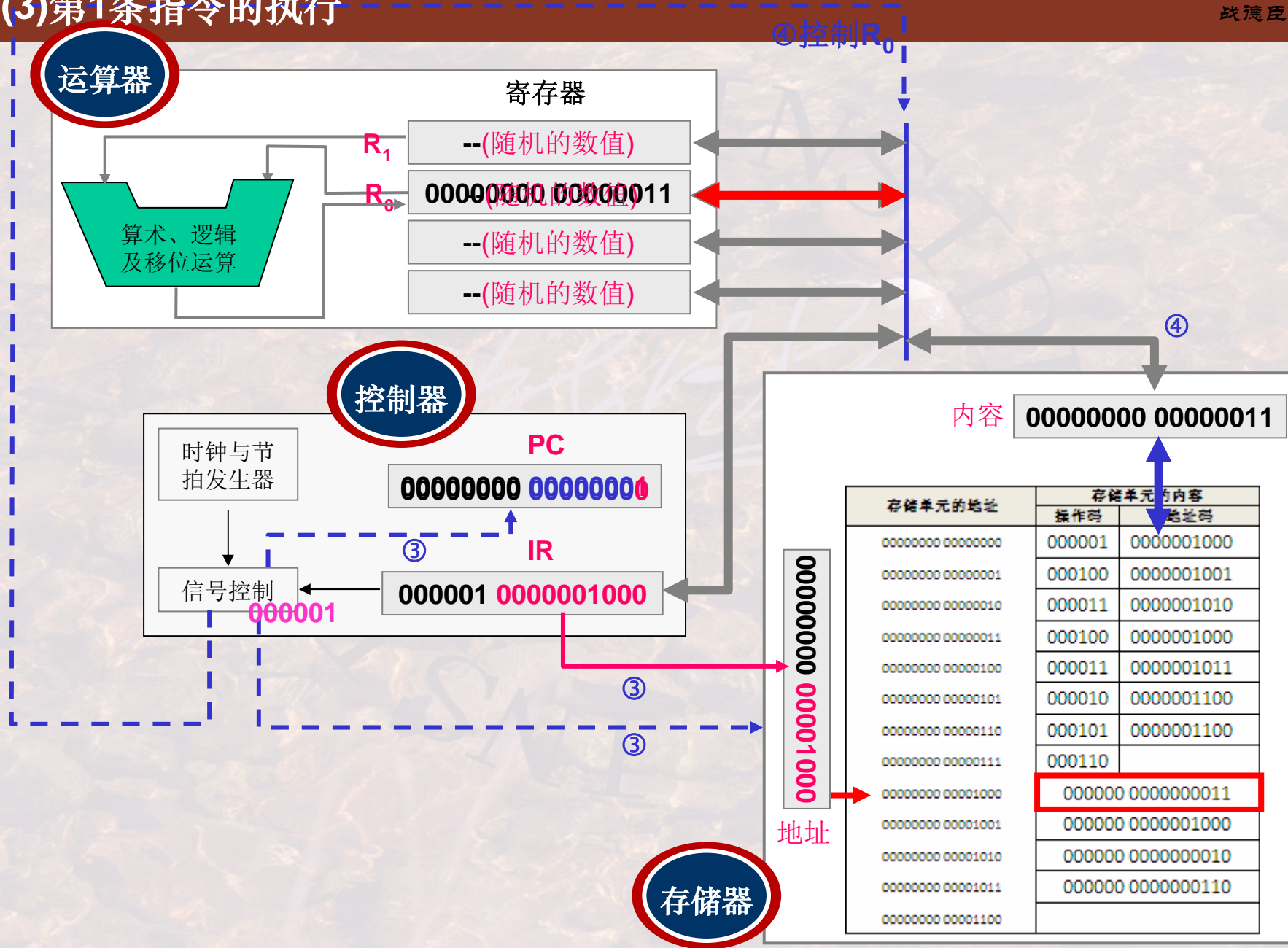


存储器



机器级程序的执行过程模拟

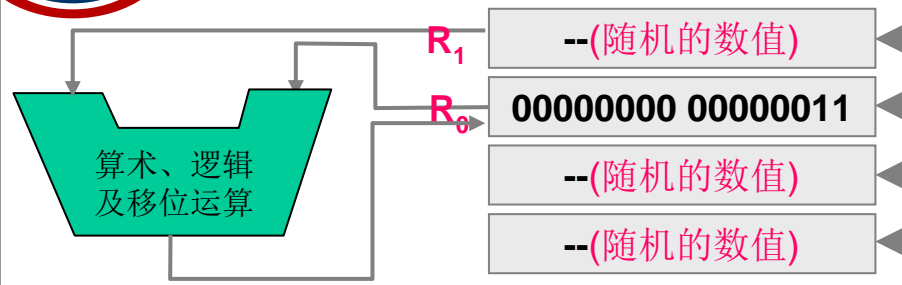
(3)第1条指令的执行



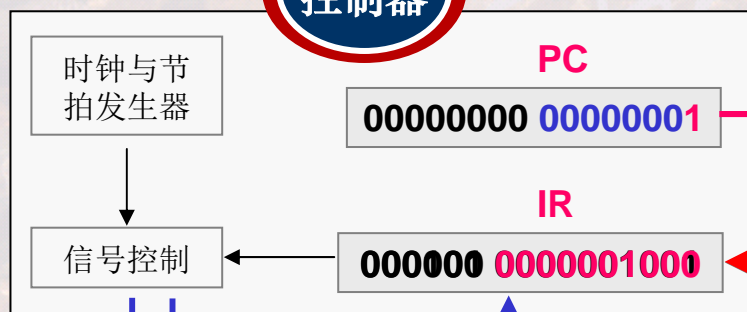
机器级程序的执行过程模拟

(4)第2条指令的读取

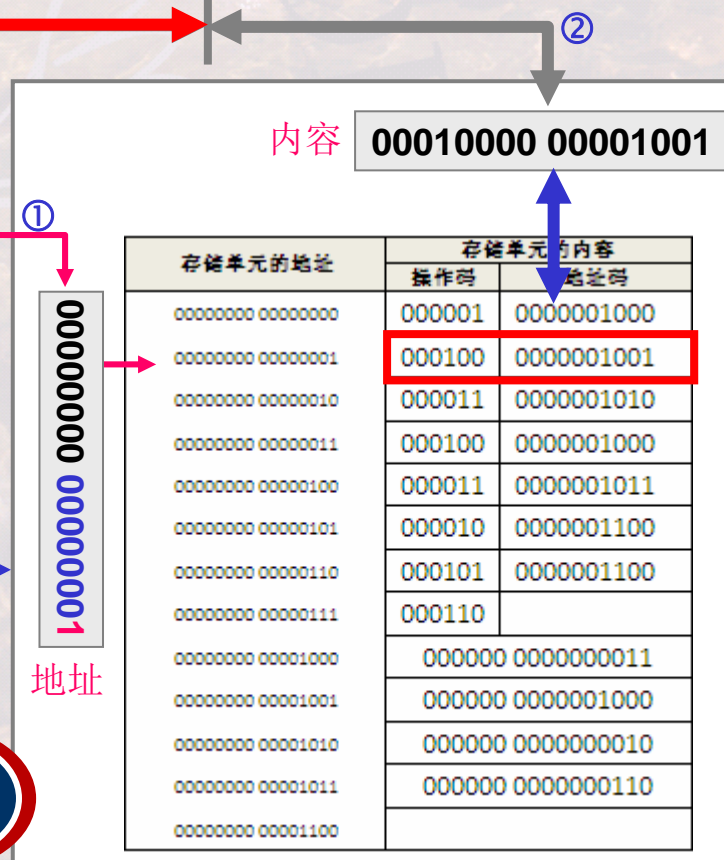
运算器



控制器

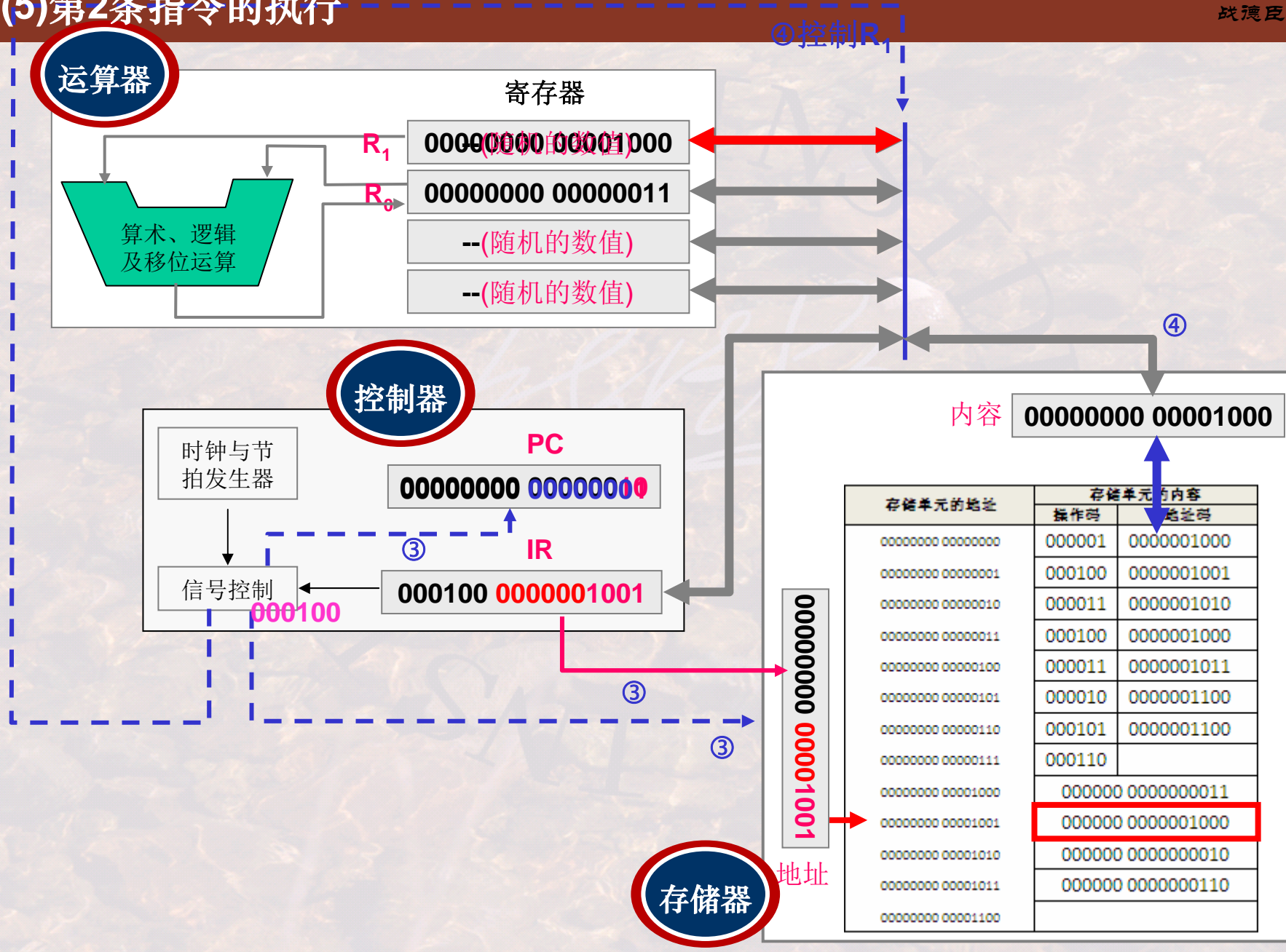


存储器



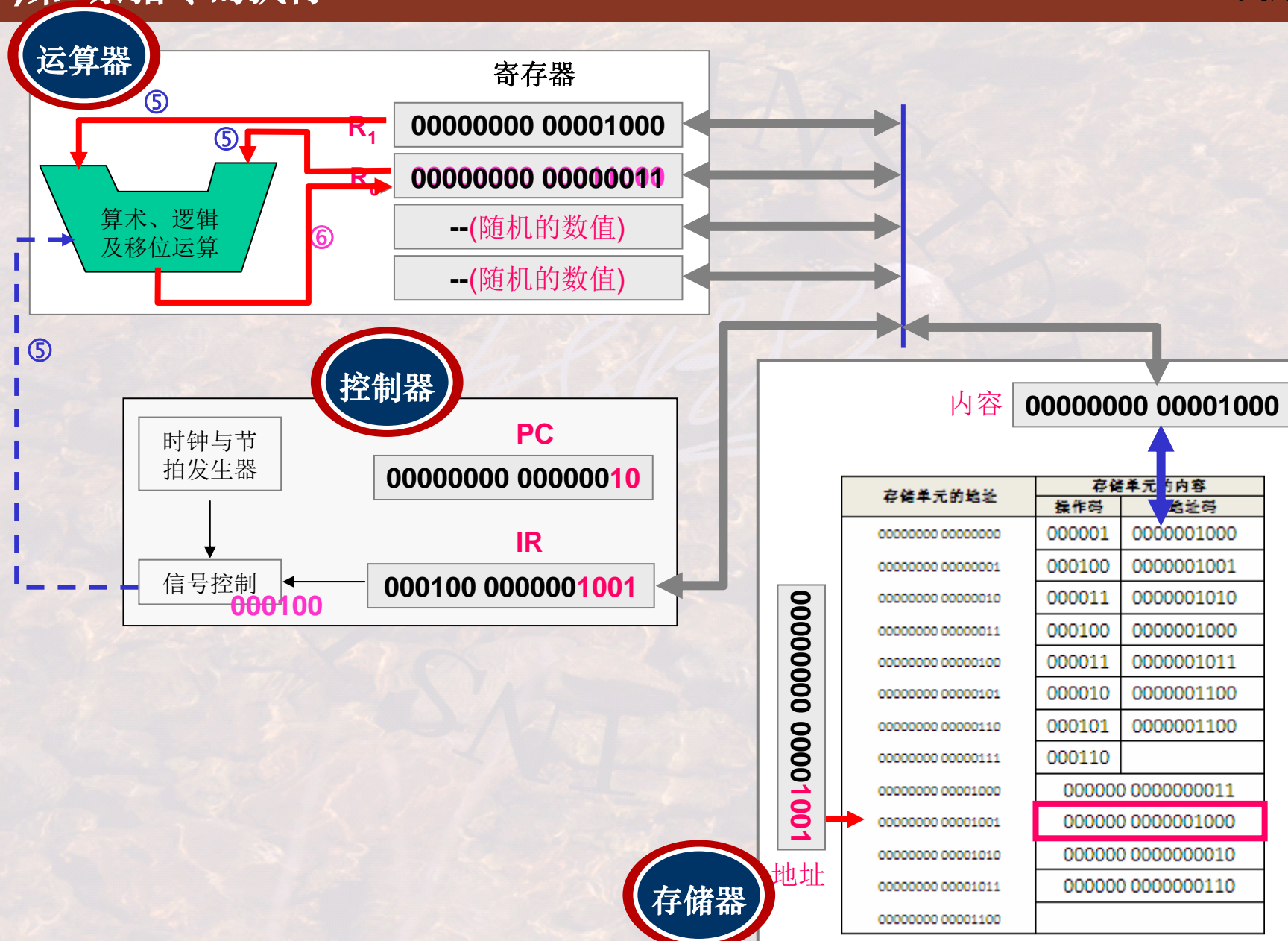
机器级程序的执行过程模拟

(5)第2条指令的执行



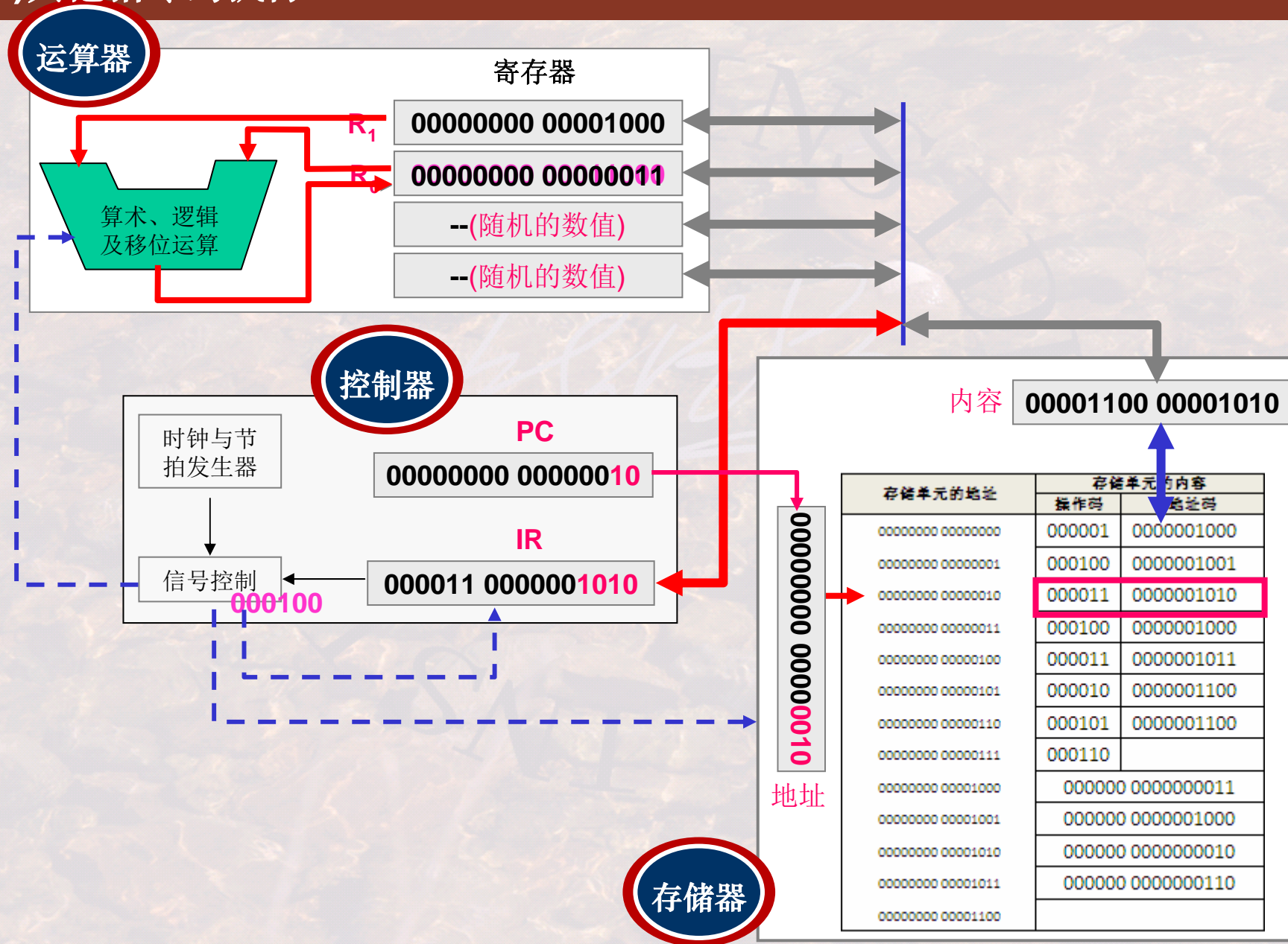
机器级程序的执行过程模拟

(5)第2条指令的执行



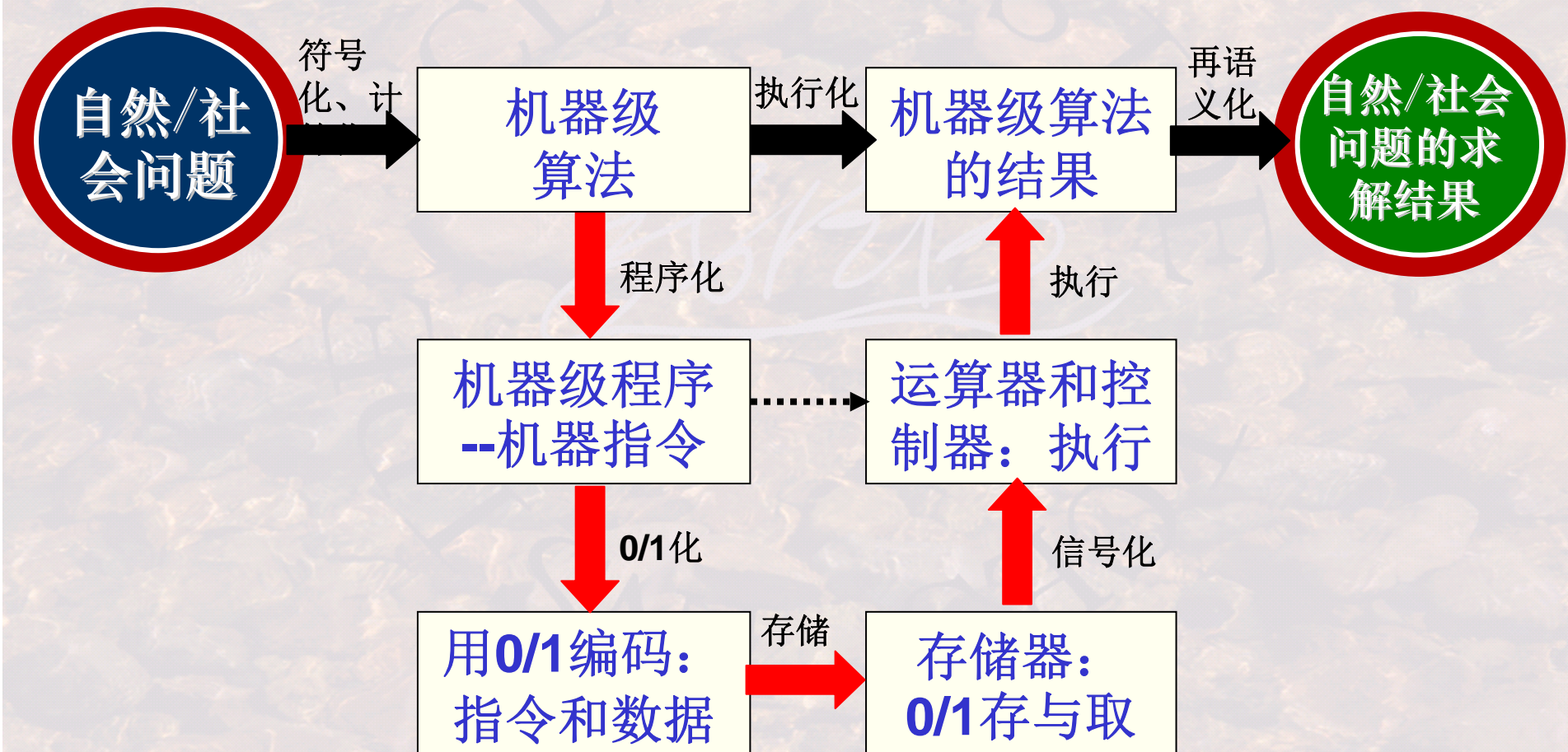
机器级程序的执行过程模拟

(6)其他指令的执行



机器级程序的执行过程模拟

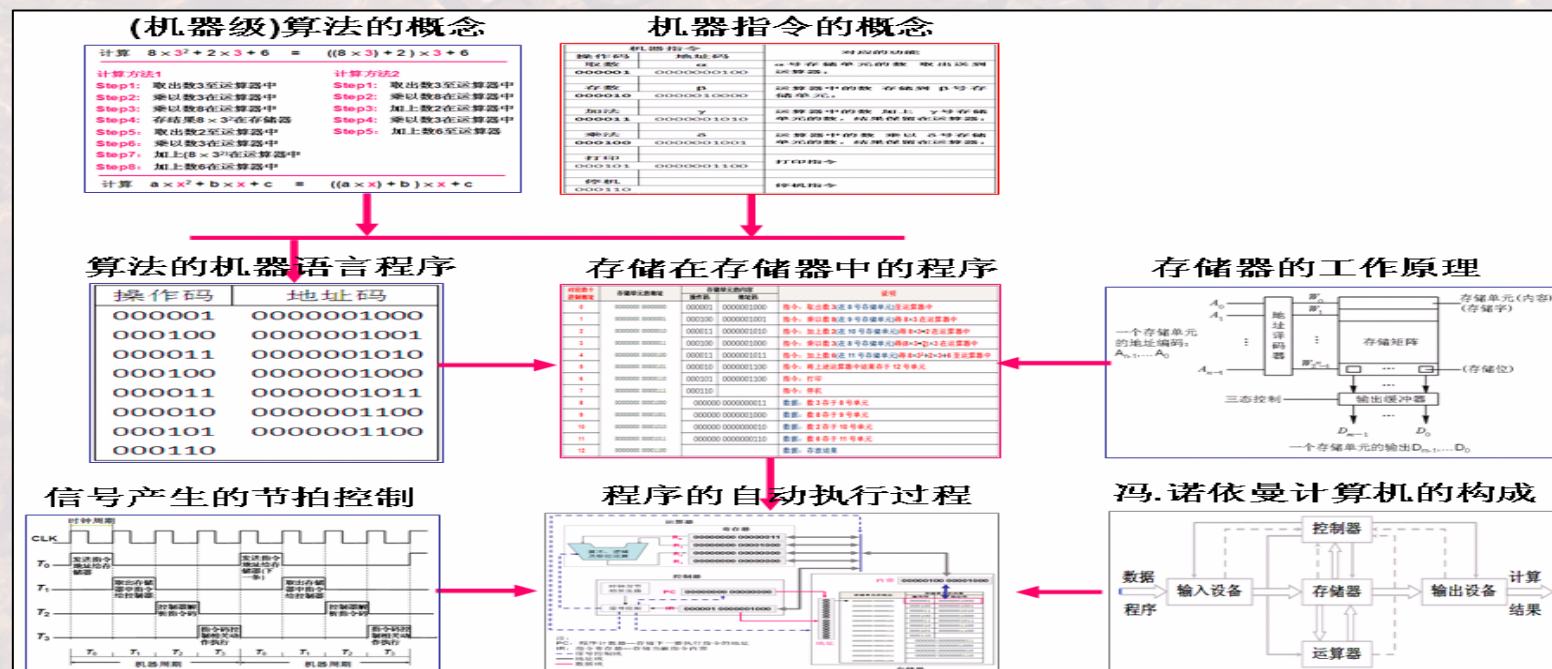
(7)小结?



机器级程序的执行过程模拟

(8)本讲总结?

基本目标: 理解程序是如何被执行的



基本思维: 机器级算法与程序 → 机器指令与指令系统 → 存储器 → 存储程序 → 运算器与控制器 → 机器级程序的执行; 算法程序化 → 程序指令化 → 指令存储化 → 执行信号化