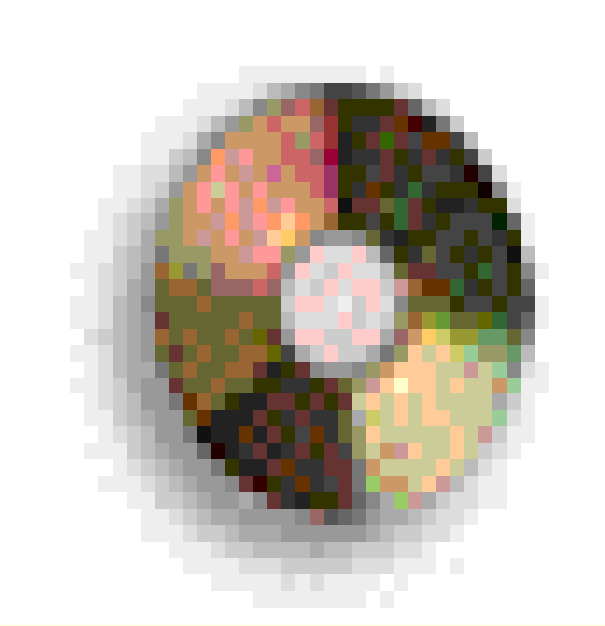
第一章　计算机基础



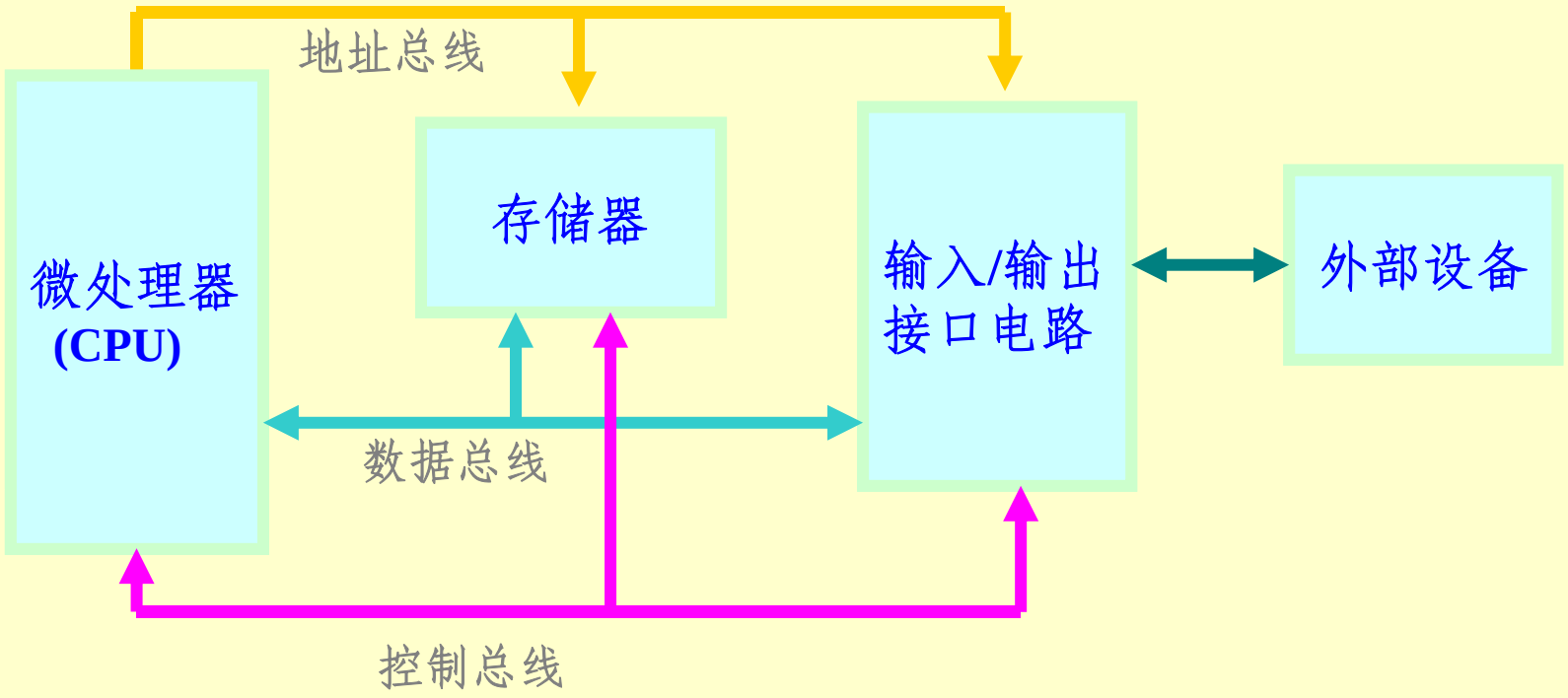
一.计算机基本结构

二、指令系统

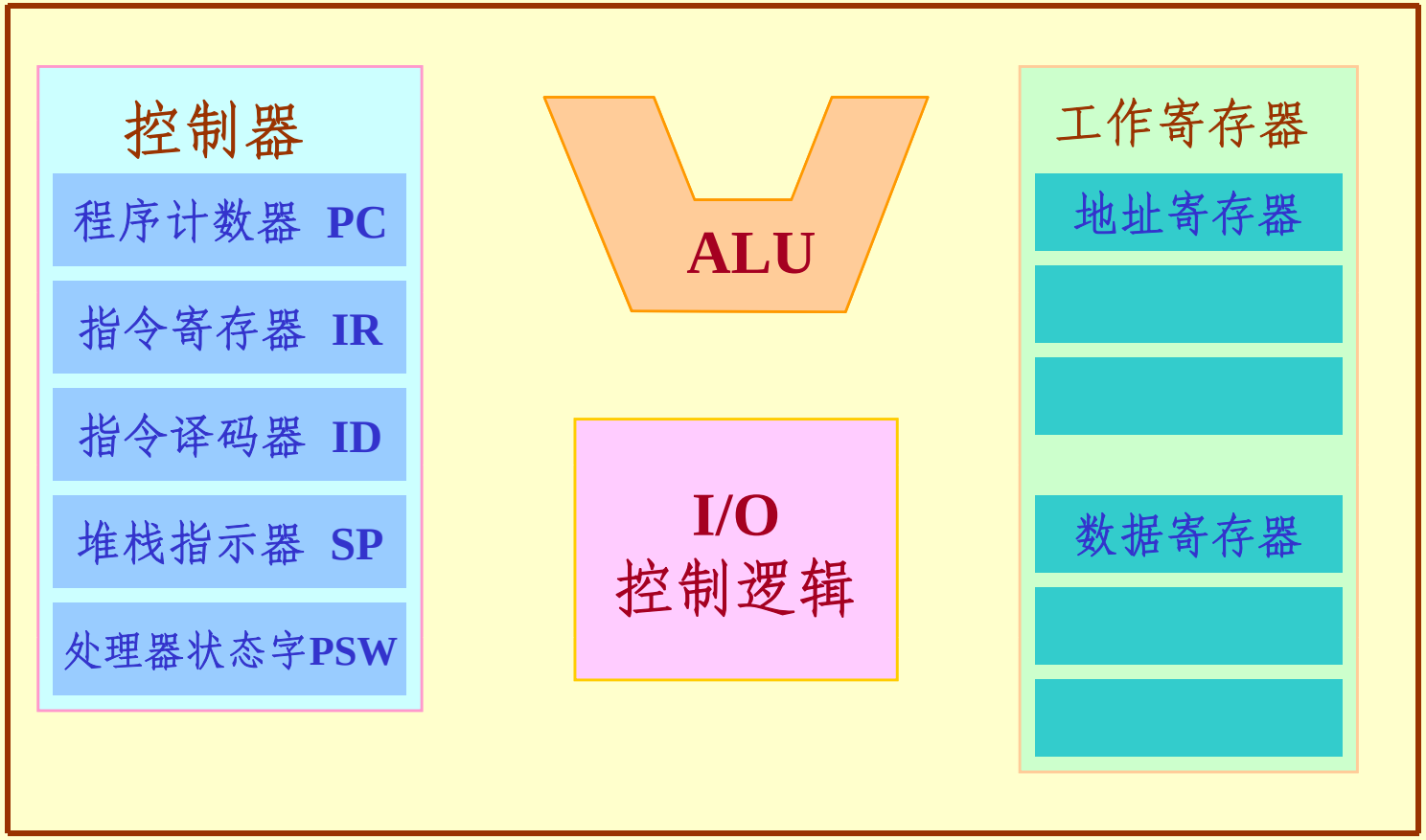
三、CPU执行过程

四、微型计算机系统

一. 计算机基本结构



1.微处理器(CPU)


控制器
程序计数器 PC
指令寄存器 IR
指令译码器 ID
堆栈指示器 SP
处理器状态字PSW
ALU
I/O
控制逻辑
工作寄存器
地址寄存器
数据寄存器

微处理器包括运算器、控制器、寄存器组三大部分,一般被集成在一个大规模集成芯片上,如8088、80x86等等,它是计算机的核心部件,具有计算、控制、数据传送、指令译码及执行等重要功能,它直接决定了计算机的主要性能.

- ALU —— 运算器的核心部件是算术逻单元ALU，所有的算术运算,逻辑运算和移位操作都是由ALU完成的.
- 控制器 —— CPU的指挥机关,完成指令的读入、寄存、译码和执行。

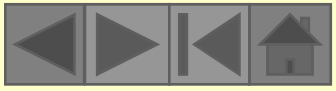
程序计数器 **PC**—— 用于保存下一条要执行的指令的地址。

指令寄存器 **IR** —— 保存从存储器中读入的当前要执行的指令。

指令译码器 **ID** —— 对指令寄存器 **IR**中保存的指令进行译码分析。

堆栈指示器 **SP** —— 对堆栈进行操作时提供地址。

处理器状态字**PSW** —— 暂存处理器当前的状态。

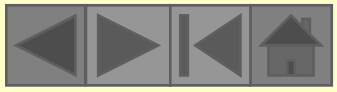


- 工作寄存器组 —— 暂存寻址和计算过程的信息.

  地址寄存器 ——地址寄存器用于操作数的寻址。

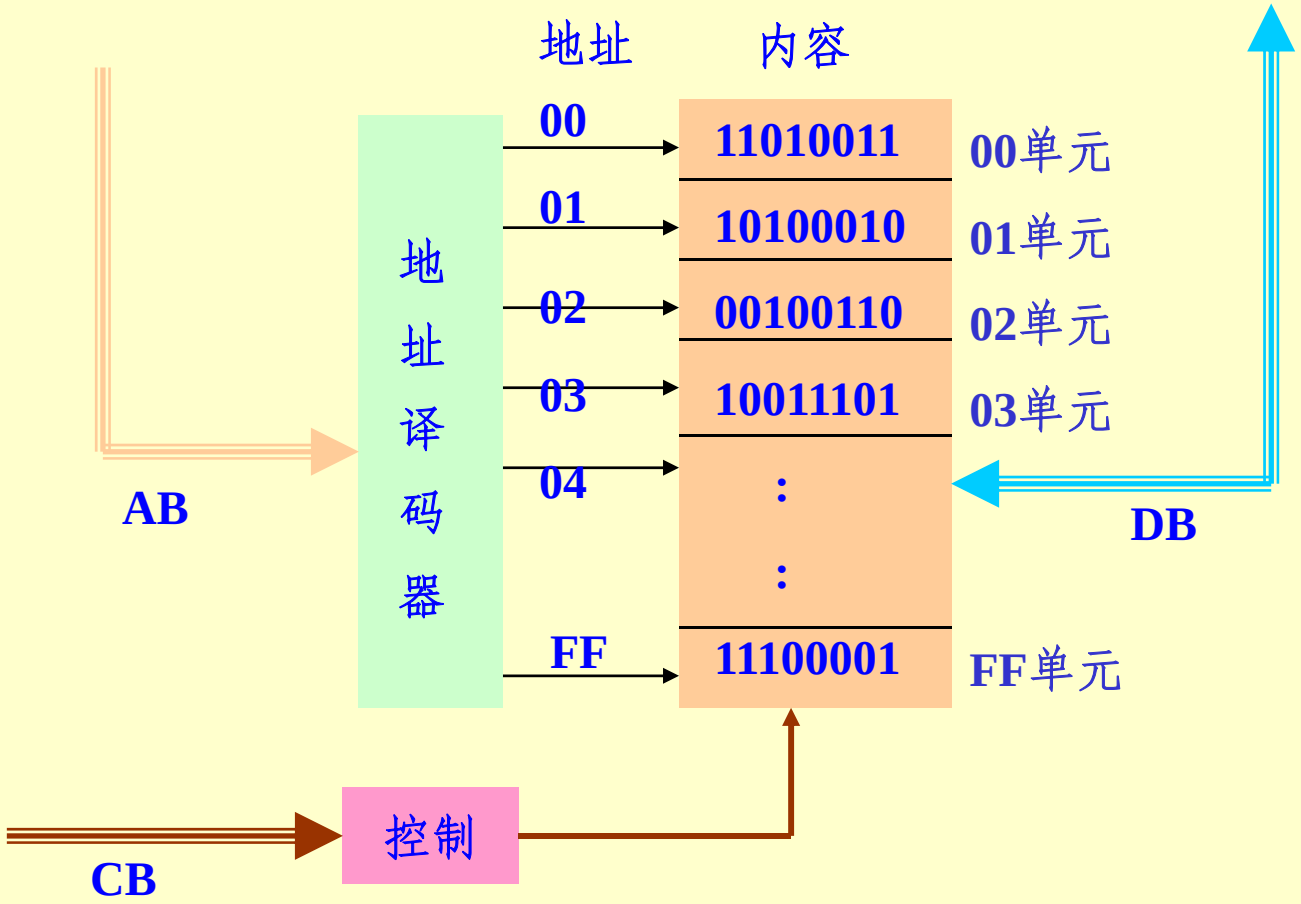
  数据寄存器 ——数据寄存器用来暂存操作数和中间运算结果。

- I/O控制逻辑 —— 包括CPU中输入/输出操作有关的逻辑，其作用是处理输入/输出的操作。



2.存储器

用于存放程序代码及有关数据.



存储器由若干存储单元、地址译码器及相应的控制电路组成。

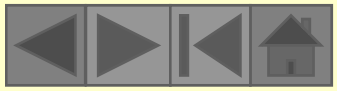
存储单元的内容：存储器由若干个单元组成，每个单元可存放 **8** 位二进制信息（通常也用两位十六进制数表示），这就是它们的内容。

存储单元的地址：为区分不同的单元，对这些单元分别编了号，这些编号即它们的地址。

存储器的读写操作：存储器中的不同存储单元，是由地址总线上送来的地址，经过存储器中的地址译码器译码，选中该单元，然后根据控制总线上的控制命令（或读或写），进行相应的读写操作。

3. 输入输出接口电路

由于外部设备如键盘、显示器、软盘、硬盘、打印机等，在数据格式、运行速度等方面与 CPU 不匹配，故在连接时，需通过 输入输出接口电路使外部设备与之相连。



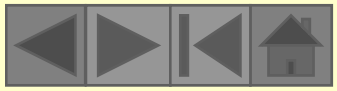
4. 总线

总线是微型计算机中模块到模块之间传输信息的通道，是各种公共信息线的集合，采用总线结构便于部件和设备的扩充。 对微机而言，总线可以分为以下四类：

片内总线——这种总线是微处理器的内总线，在微处理器内用来连接ALU、CU和寄存器组等逻辑功能单元。这种总线没有具体标准，由芯片生产厂家自己确定。

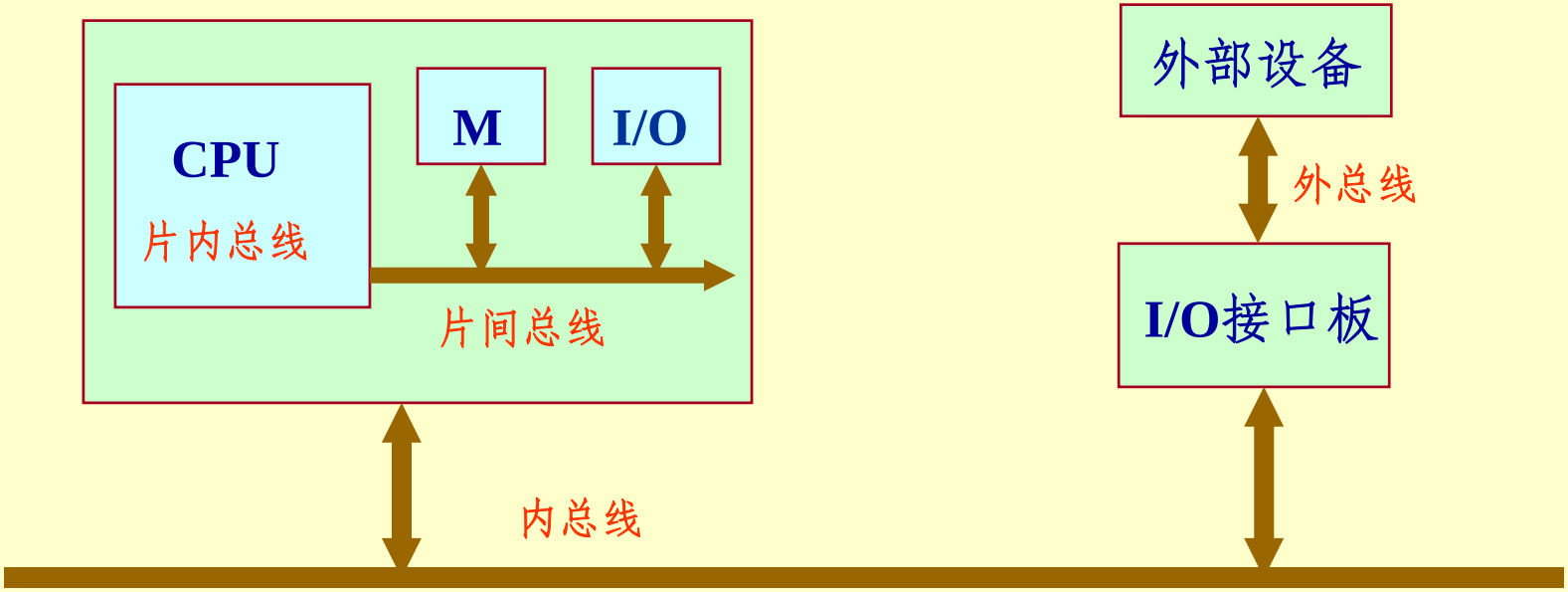
片间总线——微处理器、存储器芯片、I/O接口芯片等之间的连接总线。片间总线通常包括数据总线、地址总线和控制总线。

内总线 —— 内总线是微型计算机系统内连接各插件板的总线，内总线有不同的总线标准，如 S-100总线(IEEE-696标),STD总线,IBM-PC总线标准等，采用不同总线标准的功能板无法连接在一起。



外总线 —— 用于微型计算机系统之间或者微型计算机与外部设备之间的通信。外总线技术已经很成熟，各种应用要求皆有标准可遵循。如并行总线IEEE-488标准，串行总线RS-232标准等。

四类总线之间的关系如图所示。

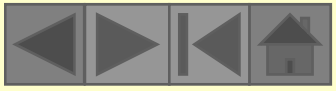


- 地址、控制、数据总线：

地址总线：用于传送 **CPU** 要访问的存储单元的地址或 **I/O** 端口地址，地址总线的位数决定了 **CPU** 可以直接寻址的地址范围。

控制总线：用来传送控制信号。

数据总线：用于 **CPU** 与存储器、 **CPU** 与外设之间传送信息。



二、指令系统

上面我们所讲述的是计算机的硬件。光有硬件，只是有了计算的可能，计算机要真正能够进行计算，还必须要有软件的配合。

例如： 4 + 5 这种简单运算,需要以下几个步骤:

① 把第一个数从它所在的存储单元中取出来,送至运算器;

② 把第二个数从它所在的存储单元中取出来,送至运算器;

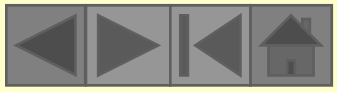
③ 相加;

④ 把加完的结果送至存储器中指定的存储单元.

所有这些取数、送数、相加、存数等等都是一种操作.

- 指令 ---- 我们把要求计算机执行的各种操作用命令的形式写下来,就是指令.

通常一条指令对应着一种基本操作,但是计算机怎么能够辨别和执行这些操作呢?这是由设计时设计人员赋予它的指令系统决定的.一个计算机能执行什么样的操作,能做多少种操作,是由设计计算机时所规定的指令系统决定的.



- 指令系统 ----- 一条指令对应着一种基本操作,计算机所能执行的全部指令,就是计算机的指令系统 . 这是计算机所固有的.

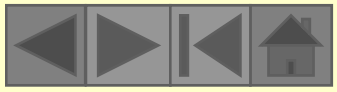
- 程序 ---- 我们在使用计算机时,必须把我们要解决的问题编成一条条指令,这些指令的集合就称为程序.

  (这些指令必须是我们所用的计算机能识别和执行的指令,也即每一条指令必须是一台特定的计算机的指令系统中具有的指令.)

  源程序 --- 用户为解决自己的问题所编的程序, 称为源程序.

- 指令形式 ---- 指令通常分成操作码(Opcode)和操作数(Operand).操作码表示计算机执行什么操作,操作数指明参加操作的数本身或操作数所在的内存中的位置.

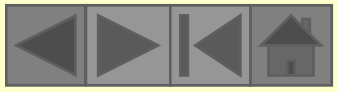
  因为计算机只认得二进制数码,所以计算机指令系统中的所有指令,都必须以二进制编码的形式来表示.如前面例子.



① 把第一个数从它所在的存储单元中取出来,送至运算器;
② 把第二个数从它所在的存储单元中取出来,送至运算器;
③ 相加;
④ 把加完的结果送至存储器中指定的存储单元.

A0 00 20	MOV AL,[2000H]
A8 1E 01 20	MOV BL,[2001H]
00 D8	ADD AL,BL
A2 00 30	MOV [3000H],AL

- 机器语言 ---- 计算机发展的初期,就是用指令的机器码直接来编制用户的源程序,这就是机器语言阶段.
- 汇编语言 ---- 由于机器码是由一连串的 0 和 1 组成的,不好记忆,容易出错,因而后来人们用一些助记符(Mnemonic)来代替操作码,如上所示.这样,每条指令有明显的特征,易于理解记忆,这便是汇编语言阶段.
- 程序的存放 ---- 要求机器能自动执行这些程序,就必须把这些程序存放到存储器的某个区域. 计算机在执行时把这些指令一条条取出来加以执行.



三、CPU执行过程

操作：　　将两个数 7 和 10 相加.

指令:　　mov al, 7

add al, 10

mov [20h], al

hlt

机器指令:　1011 0000　　B0h　　(mov al, 7)

0000 0111　　07h

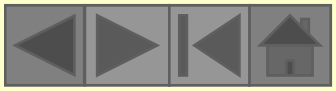
0000 0100　　04h　　(add al, 10)

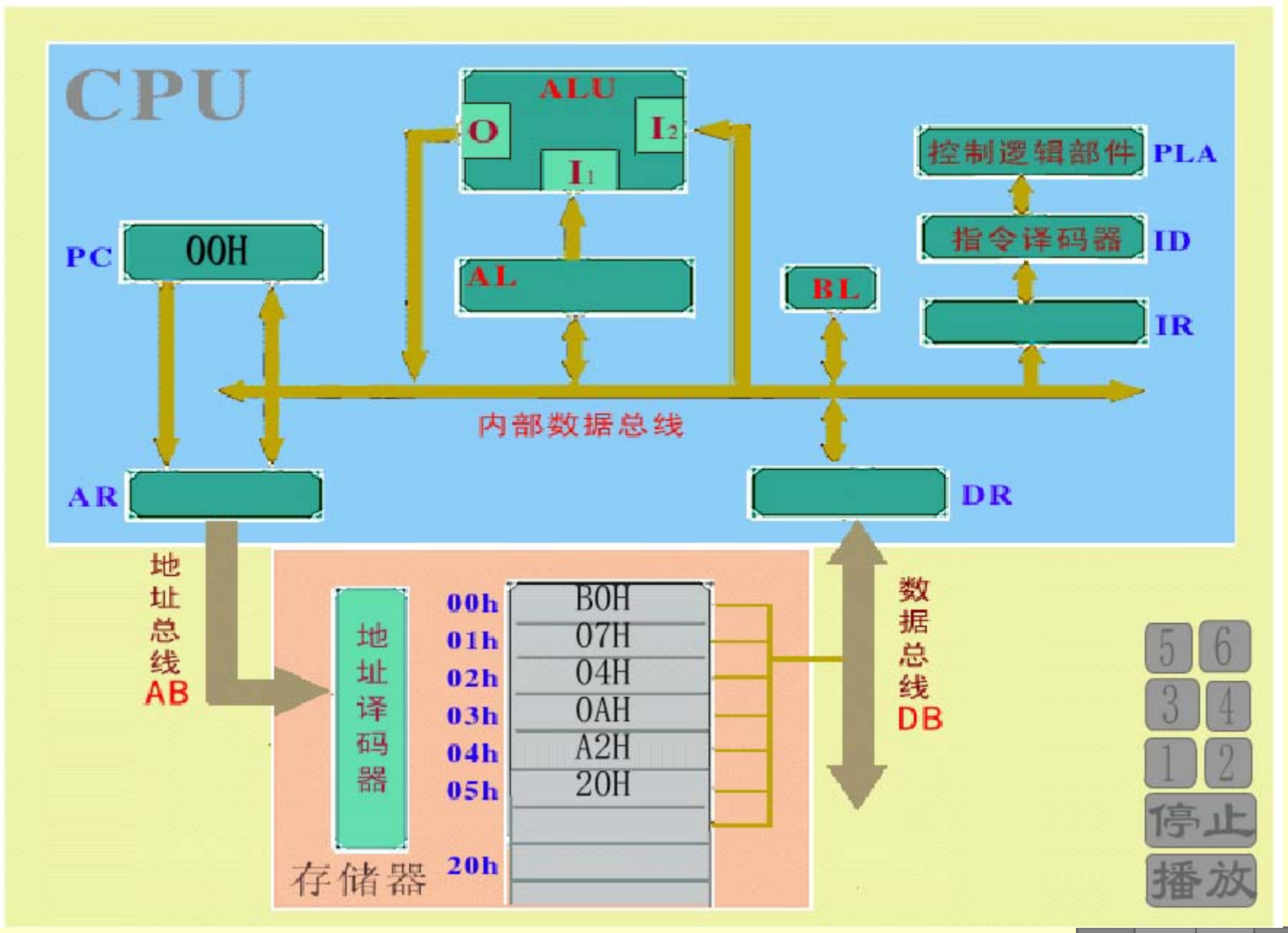
0000 1010　　0Ah

1010 0010　　A2h　　(mov [20h], al)

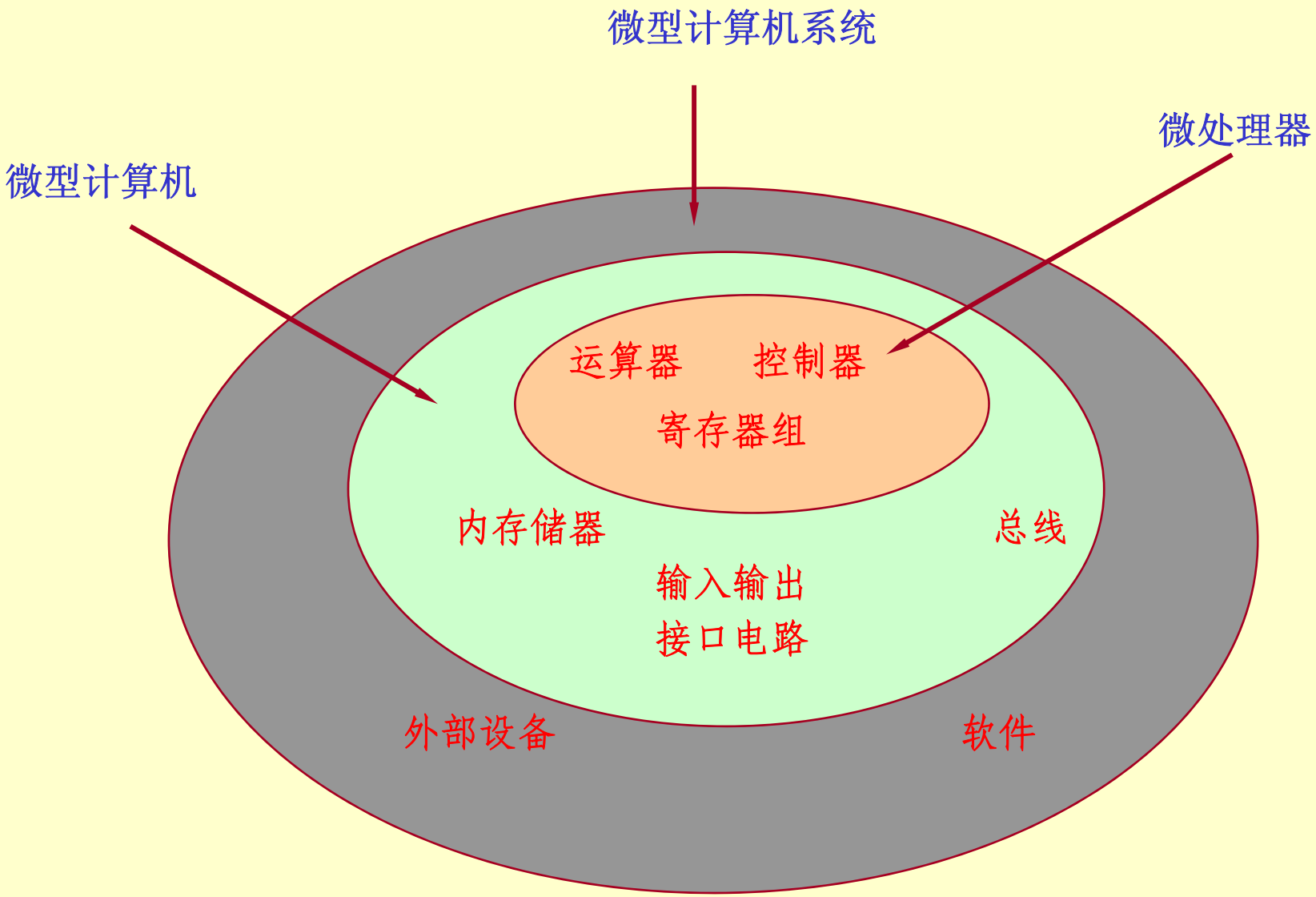
0010 0000　　20h

1111 0100　　F4h　　(hlt)



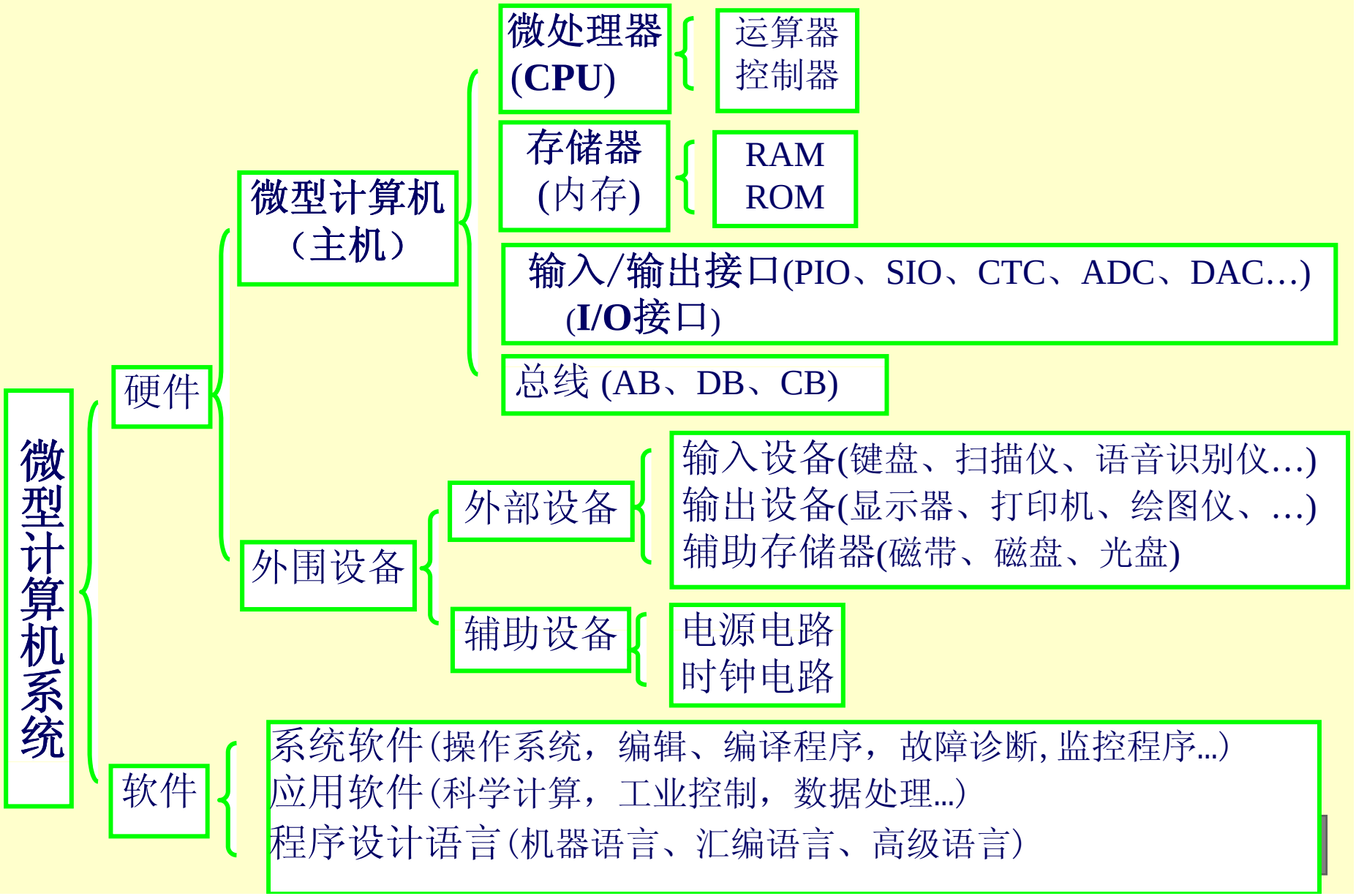

CPU
ALU
O
I2
I1
AL
BL
PC
00H
控制逻辑部件
PLA
指令译码器
ID
IR
内部数据总线
AR
DR
地址总线 AB
数据总线 DB
地址译码器
存储器
00h
01h
02h
03h
04h
05h
20h
B0H
07H
04H
0AH
A2H
20H
5
6
3
4
1
2
停止
播放

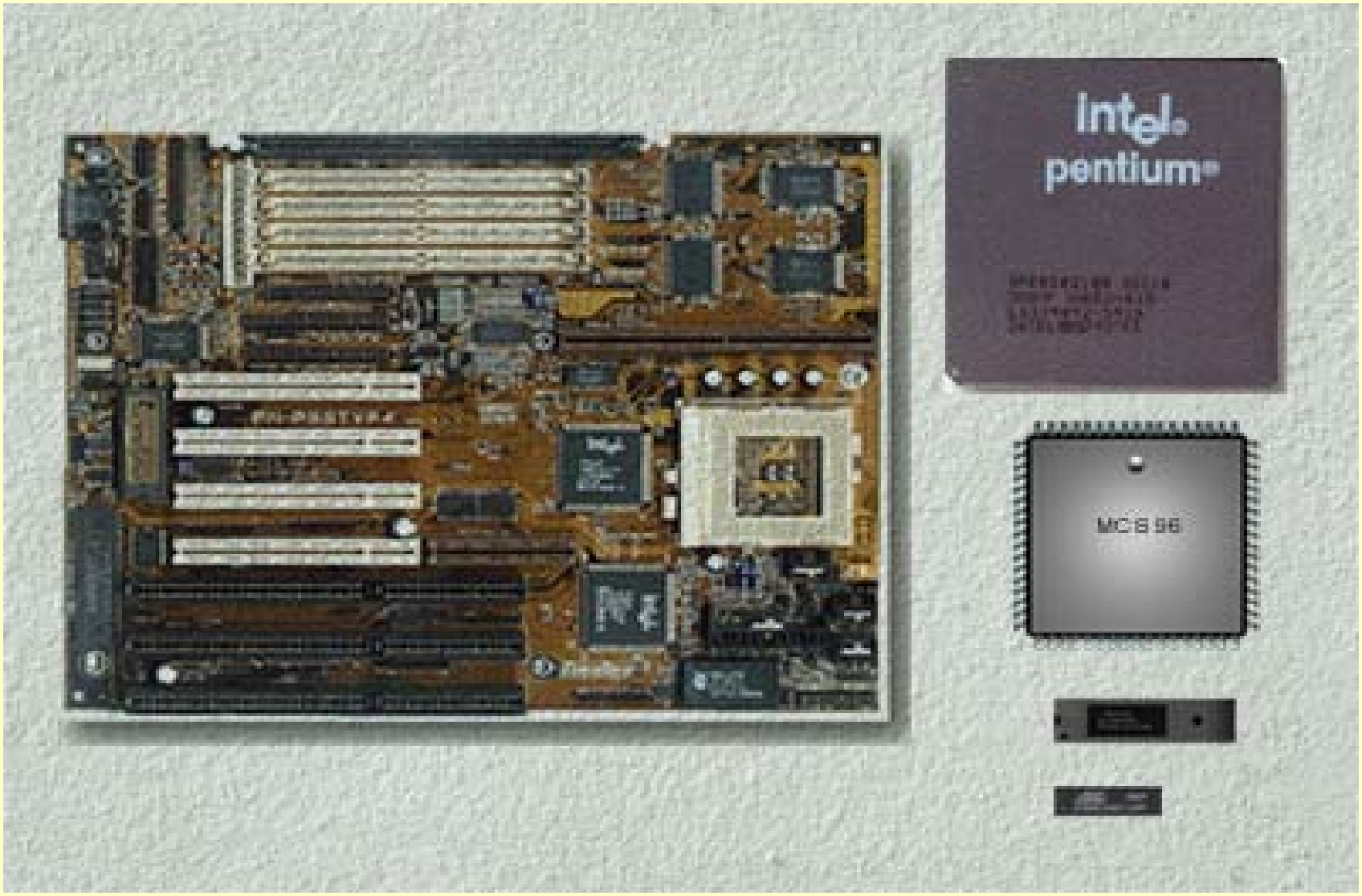
四、微型计算机系统



微型计算机系统的三个层次：

微处理器　　微型计算机　　微型计算机系统




intel
pentium
MC8 96