

A scenic landscape featuring a calm lake in the foreground, a dense forest of trees with autumn foliage in the middle ground, and rolling mountains in the background under a clear blue sky with scattered white clouds. The text '上海交通大学' is overlaid in the upper center.

上海交通大学

网络学院

# 微机原理与应用

The Principle & Application of Microcomputer

王春香 副教授

wangcx@sjtu.edu.cn

# 教材及参考书

---

## 教材

- **李继灿**主编，新编16/32位微型计算机原理及应用（第3版），清华大学出版社，2004年

## 参考书

- **周明德**编著，微型计算机系统原理及应用（第4版），清华大学出版社，2002年
- **戴梅萼**编著，微型计算机技术及应用（第3版），清华大学出版社，2003年

# 课程安排

---

 共5学分，75学时

 上课60学时：15次

上课时间：周三晚 6:30 – 9:30

上课地点：新上院314

 上机15学时：4次

上机时间：第 4, 7, 13, 15 周

周日下午 15:15 – 18:15

上机地点：新上院700号

 使用软件：MASM 611

# 考核方法

---

📖 出勤(面授、点播、下载、上机): 10%

📖 作业(4次): 20%

📖 期末考试: 70%



# 第一章 微机系统导论

## 主要内容

1.1 微机系统组成

1.2 微机硬件系统结构

1.3 微处理器组成

1.4 存储器概述

1.5 微机工作过程

1.6 微机系统的主要性能指标



# 第一章 微机系统导论

## ■ 学习要求

- 理解微机硬、软件系统的功能及其相互之间的关系。
- 理解微机硬件系统各组成部分功能与作用，掌握各种信息的不同流向。
- 理解CPU对存储器的读/写操作及其区别。
- 着重理解和熟练掌握程序执行的过程。
- 着重掌握微处理器的几个基本指标。



## ■ 第一台计算机

世界上第一台数字电子计算机1946 年在美国陆军阿伯丁弹道实验室研制成功，取名为 ENIAC (Electronic Numerical Integrator and Calculator )。

该计算机共用18000多个电子管、6000余个开关、7000个电阻、10000个电容器和50万条连线，重达30吨，占地170平方米，耗电140千瓦，运算速度5000次加法/秒。

尽管该计算机有诸多不足，如存储容量小、体积大、耗电多、可靠性差、使用不便等，但当时人们对它的速度还是相当满意。它的诞生宣布了电子计算机时代的到来。





## ■ 计算机发展经历4代

第一代电子计算机特点是使用电子管，20世纪40年末和50年代初获得重大发展；

第二代电子计算机于20世纪50年代中期问世，晶体管代替电子管，并增加了浮点运算；

1964年4月IBM360系统问世，成为使用集成电路的第三代电子计算机的著名代表；

70年以后，出现使用超大规模集成电路的第四代电子计算机，目前使用的计算机都属于第四代计算机；



## ■ 计算机发展经历4代(续)

80年代开始研制第五代计算机，目标是打破以往计算机固有的体系结构，使计算机具有像人一样的思维、推理和判断能力，向智能化发展，称为“智能计算机”。

目前，科学家们正在使计算机朝着巨型化、微型化、网络化、智能化和多功能化的方向发展。

巨型机的研制、开发和利用，代表着一个国家的经济实力和科学水平；

微型机的研制、开发和广泛应用，标志着一个国家科学普及的程度。



## ■ 微处理器发展经历6代

第1代微处理器：1972年研制的8位微处理器 Intel 8008，主要采用工艺简单、速度较低的P沟道MOS电路；

第2代微处理器：1973年研制的，主要采用速度较快的N沟道MOS技术的8位微处理器。代表产品有Intel的Intel 8085、Motorola的M6800、Zilog的Z80等；

第3代微处理器：1978年研制的，主要采用H-MOS新工艺的16位微处理器。其典型产品是Intel的Intel 8086；



## ■ 微处理器发展经历6代(续)

第4代微处理器：1985年起采用超大规模集成电路的32位微处理器，典型产品有Intel的Intel 80386、Zilog的Z8000、惠普公司的HP-32等；

1993年Intel公司推出第五代32位微处理器芯片Pentium，外部数据总线为64位，工作频率为66 - 200MHz；

1998年Intel公司推出Pentium II、Celeron，后来又推出Pentium III。第六代都是更先进的32位高档微处理器，工作频率为300 - 860 MHz，主要用于高档微机或服务器。



# 1.1 微机系统组成

## 1.1.1 基本定义1

**微处理器**( $\mu$  P,MP-microprocessor): 由一片或几片大规模集成电路组成的, 具有运算器和控制器功能的中央处理器, 也称为微处理机。

大、中、小型**中央处理器**(CPU-Central Processing Unit),  
**微处理器** (MPU-Microprocessing Unit)



# 英特尔高性能家用台式机微处理器





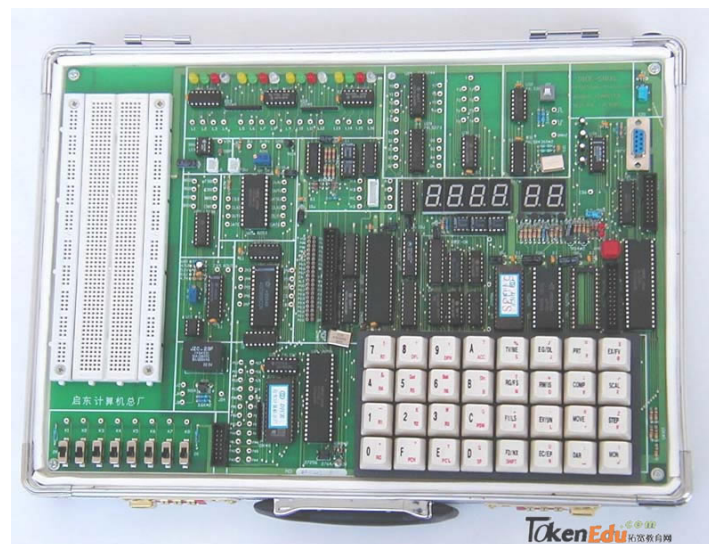
# 1.1 微机系统组成



## 1.1.1 基本定义2

**微型计算机(microcomputer):**  
简称  $\mu$ C 或 MC, 以微处理器为核心, 配上存储器、输入输出接口电路及系统总线所组成的计算机, 又称为**主机**或微电脑。

把微处理器、存储器、输入输出接口电路组装在**一块或多块**电路板上或集成在单片**芯片**上, 分别称为**单板机**、**多板机**或**单片微型计算机**。





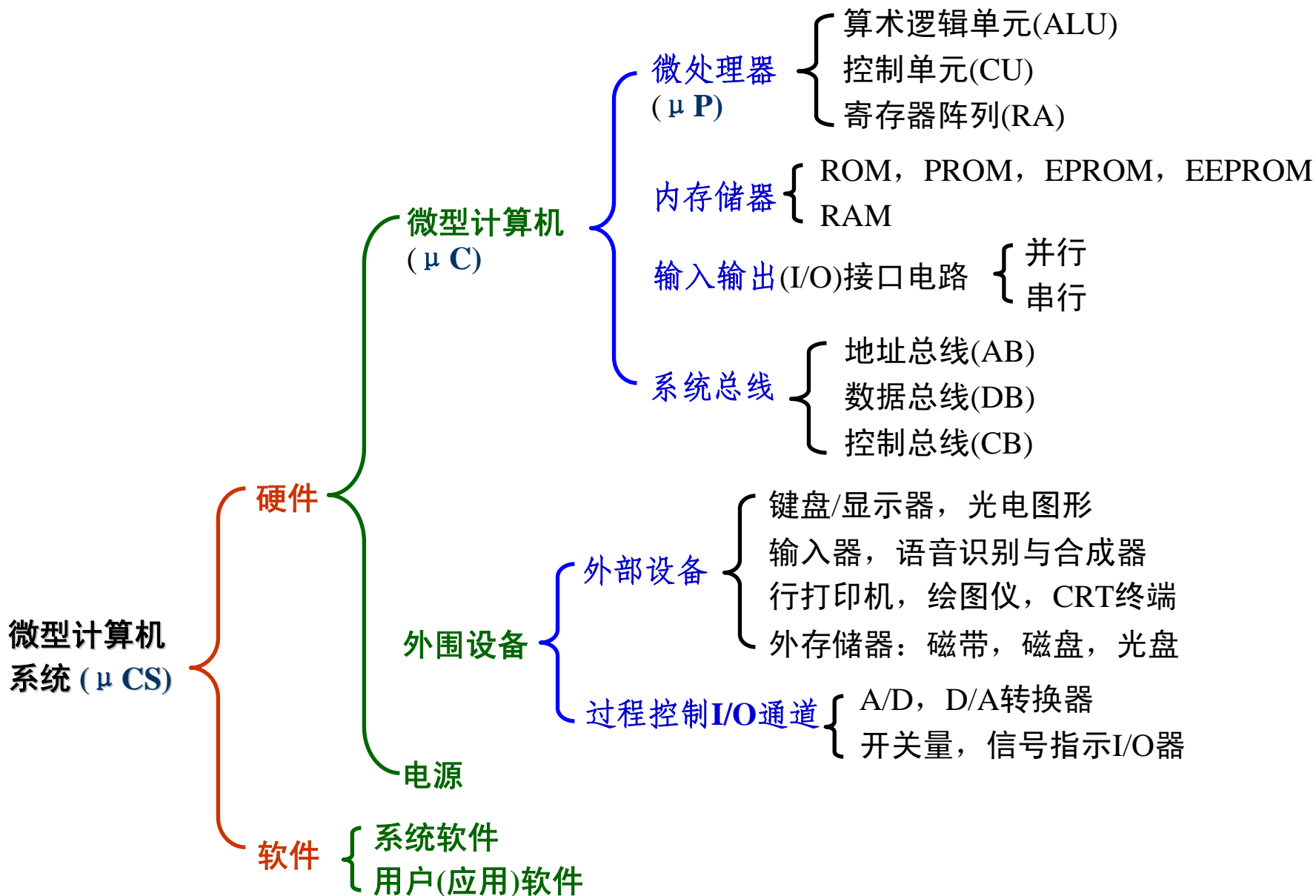
# 1.1 微机系统组成

## 1.1.1 基本定义3

微型计算机系统(microcomputer system): 简称  $\mu$  CS 或 MCS, 指以微型计算机为中心, 配上相应的外围设备、电源和辅助电路(通称**硬件**), 以及指挥微型机工作的系统**软件**所构成的系统。



# $\mu$ CS、 $\mu$ C、 $\mu$ P相互关系





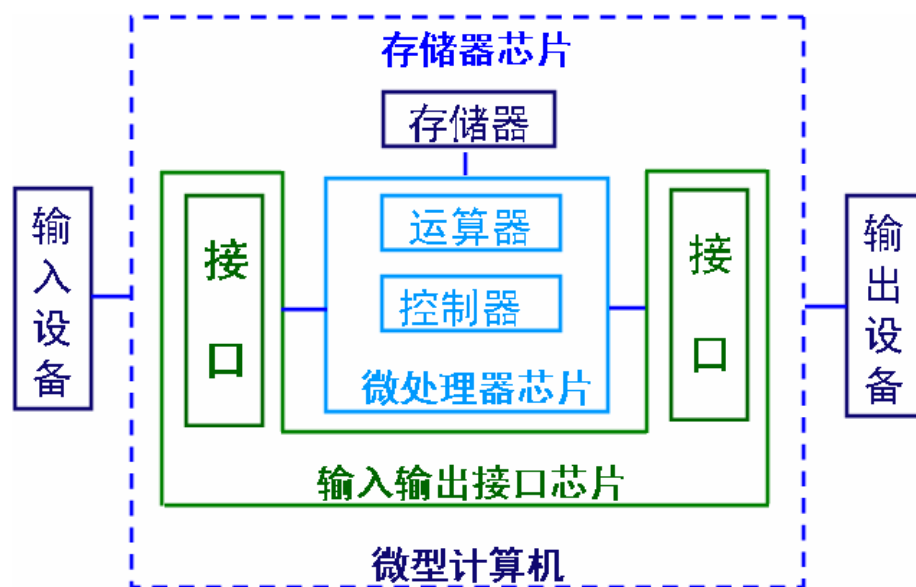
# 1.1 微机系统组成

## □ 微机硬件系统组成

**微处理器：**微机的计算、控制中心，用来实现算术、逻辑运算以及其他操作，并对全机进行控制。

**存储器(主存或内存)：**用来存储可以供微处理器直接运行的程序或处理的数据。

**输入/输出(I/O)：**接口芯片是微处理器与外部输入/输出设备之间的接口。



# 1.1 微机系统组成



## □ 微机硬件系统组成

目前，最流行的实际微机硬件系统组成：

- **主机板：**包括CPU、CPU外围芯片组、主存储器RAM、BIOS芯片与总线插槽
- **外设接口卡：**如显卡、声卡、网卡
- **外部设备：**如硬盘、光驱、显示器、打印机、键盘、调制解调器与鼠标
- **电源**等部件。



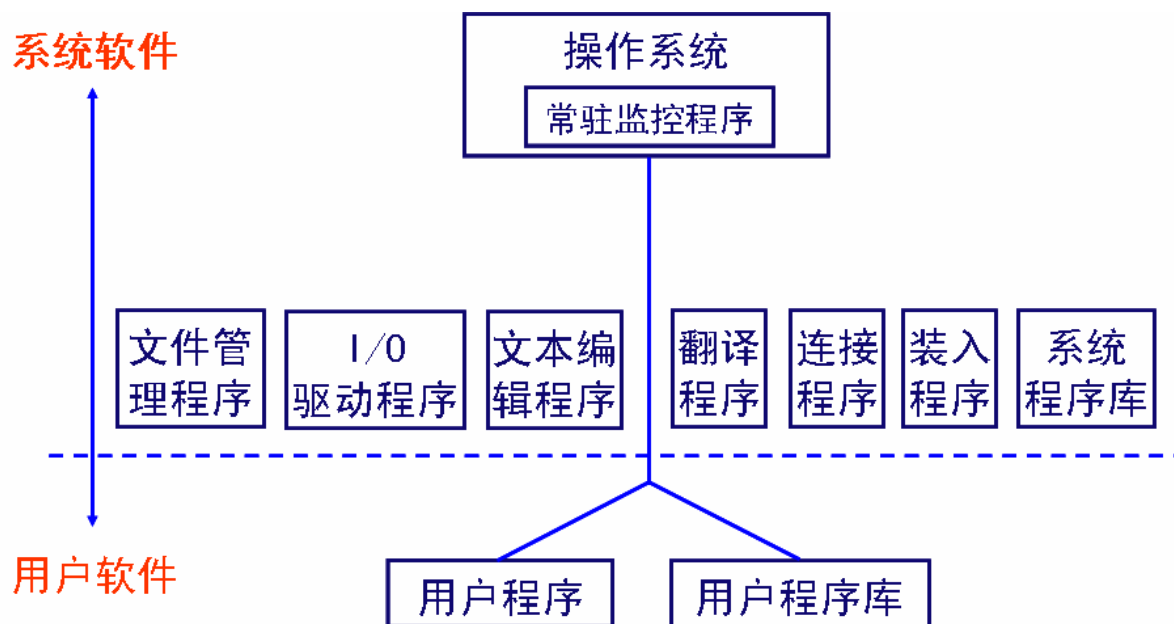
# 1.1 微机系统组成

## □ 软件的分级结构

计算机软件通常分为**两大类**：系统软件 and 用户软件。

**系统软件**是指不需要用户干预的能生成、准备和执行其他程序所需的一组程序。

**用户软件**是各用户为解题或实现检测与实时控制等不同任务所编制的应用程序，它也称为应用软件。





# 1.1 微机系统组成



## □ 软件的分级结构

操作系统是一套复杂的系统程序，用于提供人机接口和管理、调度计算机的所有硬件与软件资源。

它所包含的系统程序的具体分类尚不统一。其中，最为重要的核心部分是常驻监控程序。

计算机开机后，常驻监控程序始终存放在内存中，它通过接收用户命令，并启动操作系统执行相应的操作。

# 1.1 微机系统组成



## □ 软件的分级结构

操作系统包括I/O驱动程序和文件管理程序。

**I/O驱动程序：**用于执行I/O操作；

**文件管理程序：**用于管理存放在外存(或海量存储器)中的大量数据集合。

当用户程序或其他系统程序需要使用I/O设备时，通常并不是由该程序执行操作,而是由操作系统利用I/O驱动程序来执行任务。**文件管理程序与I/O驱动程序配合使用**，用于文件的存取、复制和其他处理。

此外，系统软件还可包括各种高级语言翻译程序、汇编程序、文本编辑程序以及辅助编写其他程序的程序。

# 1.1 微机系统组成



**应当指出**，硬件系统和软件系统是相辅相成的，共同构成微型计算机系统，缺一不可。

现代的计算机硬件系统和软件系统之间的分界线并不明显，总的趋势是两者统一融合，在发展上互相促进。

人是通过软件系统与硬件打交道。通常，由人使用程序设计语言编制应用程序，在系统软件的干预下使用硬件系统。

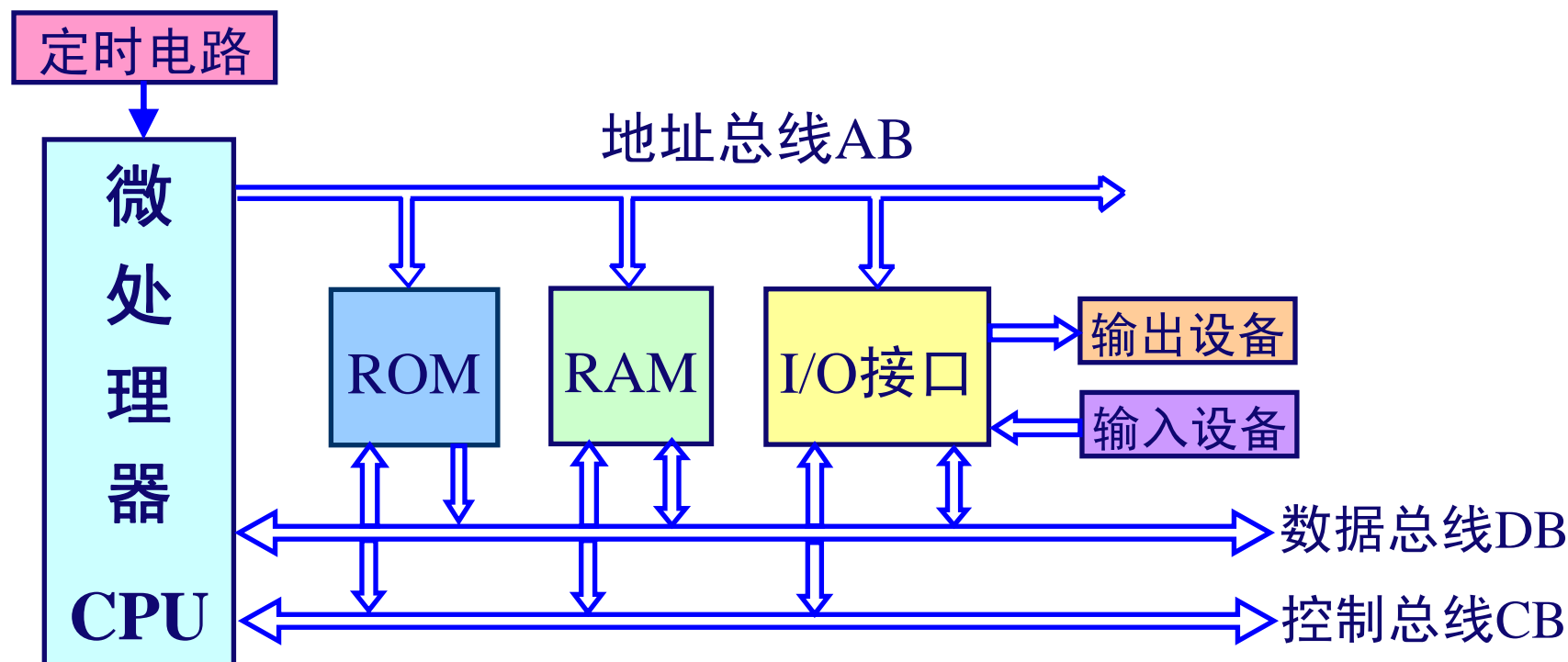


# 1.2 微机硬件系统结构

## □ 典型的微机硬件系统结构

**硬件系统结构：**按总体布局的设计要求将各部件构成某个系统的连接方式。

用**总线**将各个部件连接起来，面向系统的总线结构。



# 1.2 微机硬件系统结构



**系统总线：**用来传送信息的公共导线，可以是带状的扁平电缆线，也可是印刷电路板上的一层极薄的金属连线。

所有信息都是通过总线在CPU、M、I/O中间传送。

根据传送信息的内容与作用，总线分为3类：

数据总线(Data Bus, DB)

地址总线(Address Bus, AB)

控制总线(Control Bus, CB)

**总线结构：**系统中各部件均挂在总线上，可使微机系统的结构比较简单，易于维护，并有更大的灵活性和更好的可扩展性。

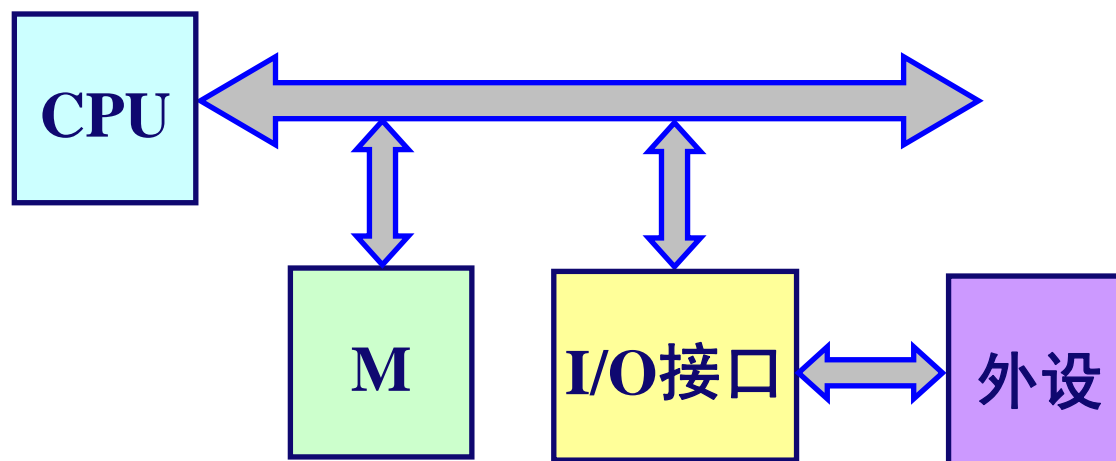
# 1.2 微机硬件系统结构



## □ 微机总线结构

根据总线结构组织方式不同，分为：  
单总线、双总线、双重总线。

- ◆ 单总线结构：M和I/O使用同一组信息通路，采取分时操作。  
结构简单，成本低。中低档微机采用此结构。



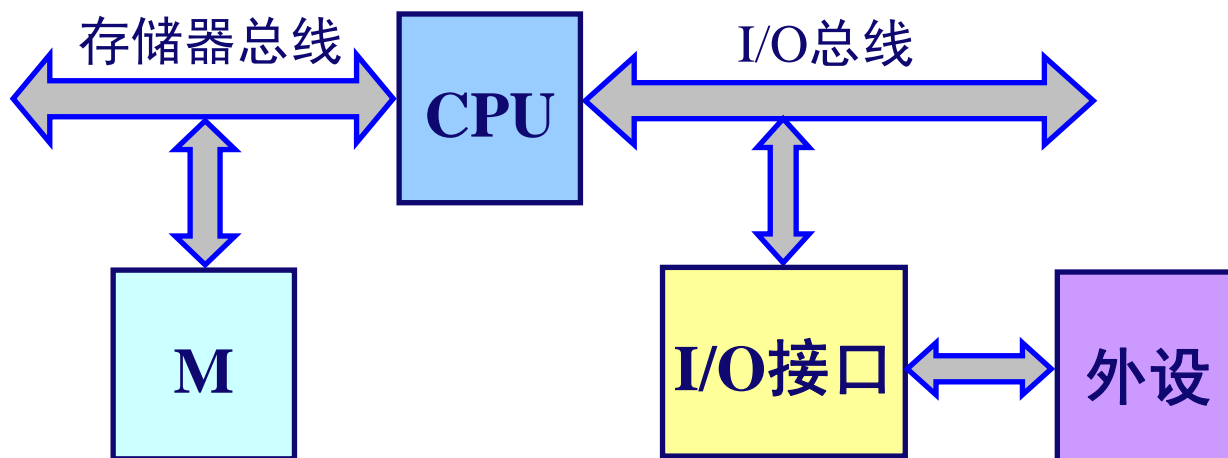


# 1.2 微机硬件系统结构



## □ 微机总线结构

- ◆ 双总线结构：M和I/O各有一组连通CPU的总线，可同时操作。  
拓宽了总线宽度，提高了传输效率，高档微机采用此结构。  
CPU负担加重，可采用专门处理芯片(智能I/O接口)减轻负担。

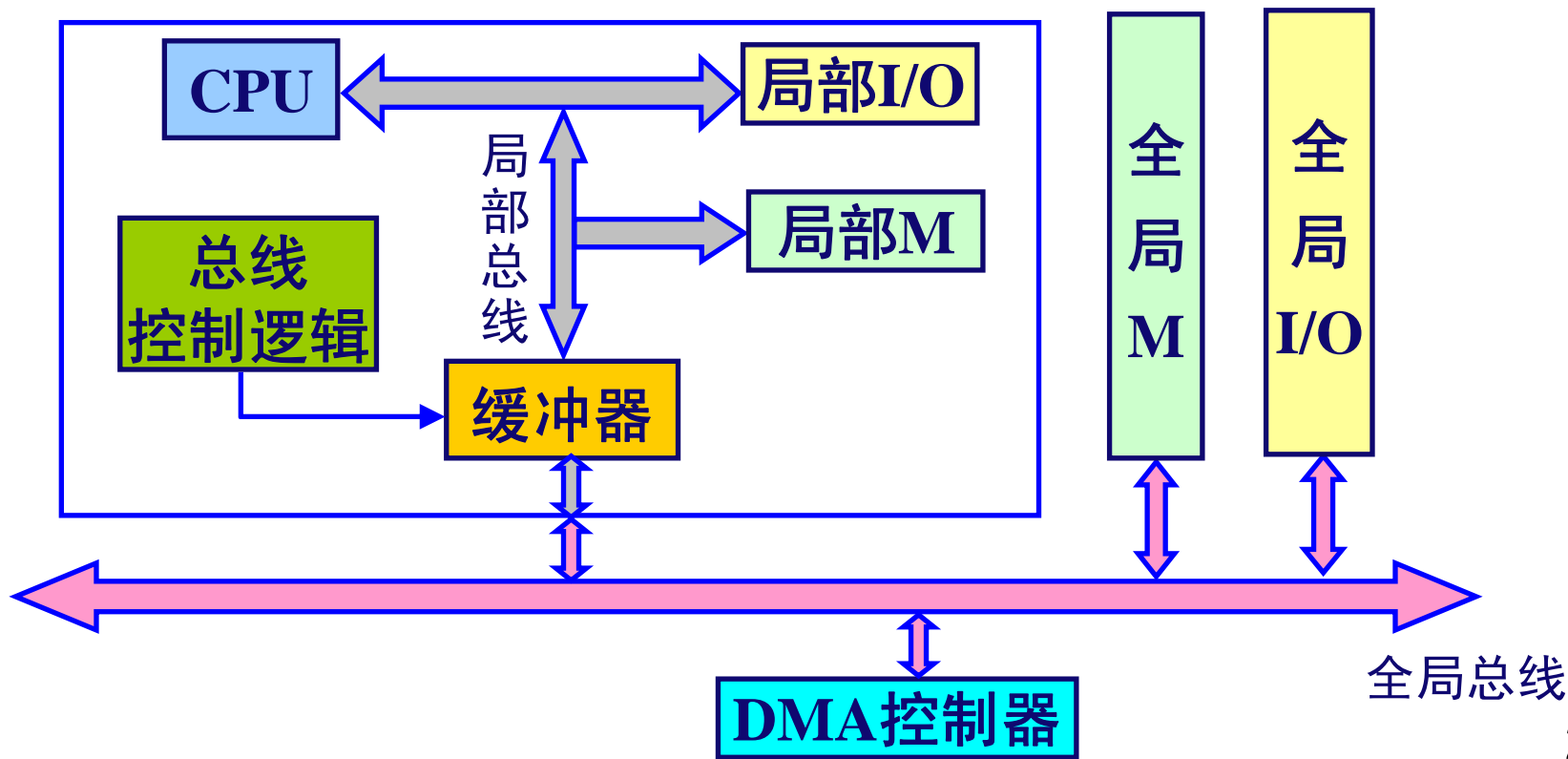




# 1.2 微机硬件系统结构

## □ 微机总线结构

- ◆ 双重总线结构：具有**局部**总线与**全局**总线，并行操作。  
提高了数据处理和传输效率，高档微机和工作站采用此结构。

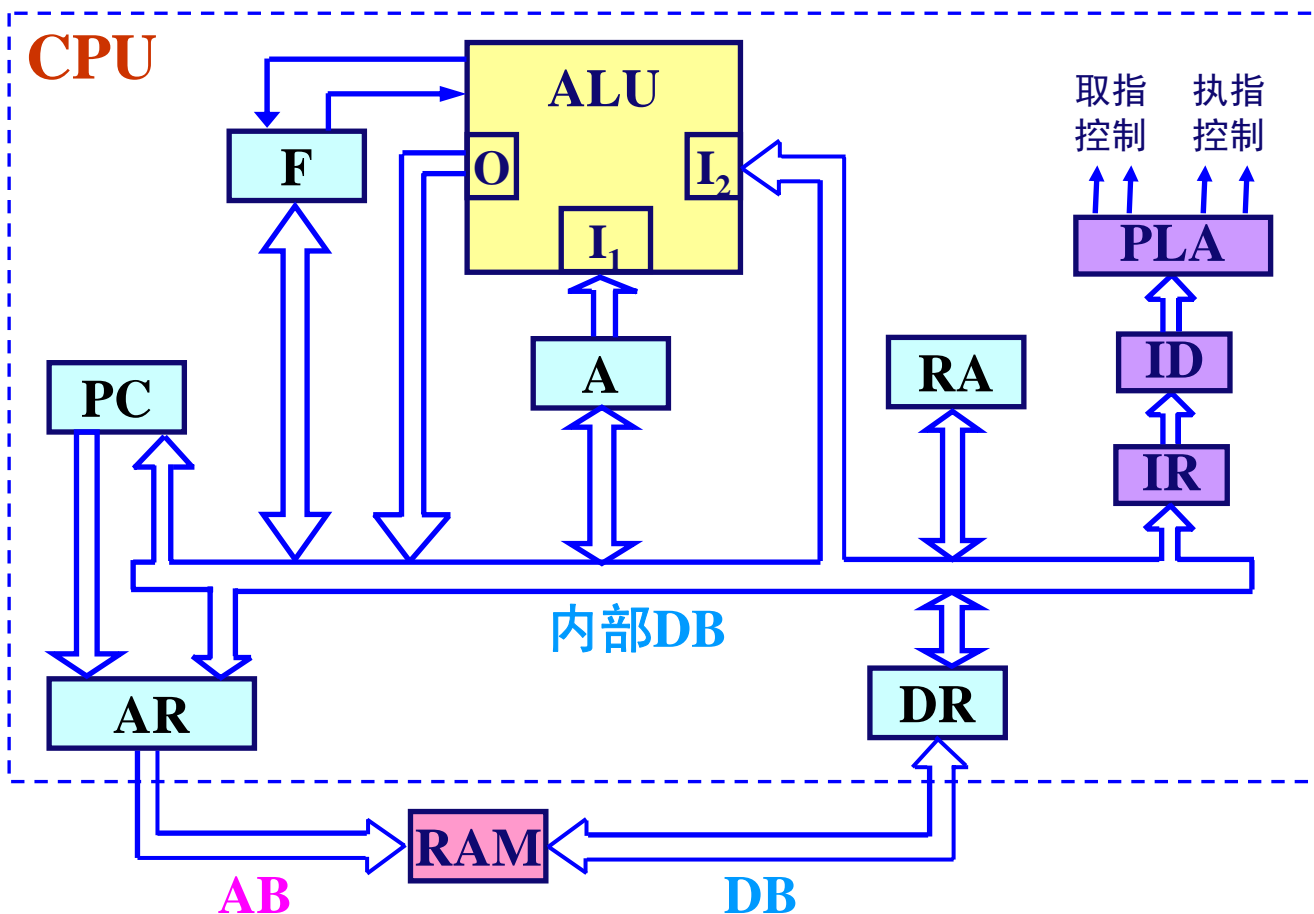


# 1.3 微处理器组成



## □ 微处理器结构

由运算器、控制器、内部寄存器阵列三部分组成。



# 1.3 微处理器组成



## 1.3.1 运算器

又称为算术逻辑单元 (Arithmetic and Logic Unit, ALU)

用来进行算术或逻辑运算，以及位移循环等操作。

参加运算的两个操作数：

- 一个来自累加器A (Accumulator)
- 另一个来自内部数据总线：
  - 数据寄存器DR (Data Register)
  - 寄存器阵列RA

运算结果送回至累加器A暂存

# 1.3 微处理器组成



## 1.3.2 控制器

### 1. 指令寄存器 (Instruction Register, IR)

存放从存储器取出的**将要执行的指令** (操作码)。

### 2. 指令译码器 (Instruction Decoder, ID)

对指令寄存器IR中的指令译码，确定该指令**执行什么操作**。

### 3. 可编程逻辑阵列 (Programmable Logic Array, PLA)

也称为定时与控制电路，产生取指令和执行指令所需的各种**微操作控制信号**。

# 1.3 微处理器组成



## 1.3.3 内部寄存器

包括若干个**功能不同**的寄存器或寄存器组。

### 1. 累加器 A (Accumulator)

算术逻辑运算时 { 运算前：保存一个操作数  
运算后：保存结果

### 2. 数据寄存器 DR (Data Register)

**暂存数据或指令**。从M读出时，

{ 若读出的是**指令**：经DR暂存的指令，经内部DB送到IR  
若读出的是**数据**：经内部DB，送到相关寄存器或运算器



# 1.3 微处理器组成



## 1.3.3 内部寄存器

### 3. 程序计数器 PC (Program Counter)

**存放正待取出的指令的地址。**根据PC中的指令地址，准备从M中取出将要执行的指令。

程序按顺序逐条执行，**PC具有自动加1的功能。**

### 4. 标志寄存器 F (Flag Register)

**寄存执行指令时所产生的结果或状态的标志信号。**

标志位的具体设置与功能和微处理器型号有关。

根据检测有关的标志位是0或1，可按不同条件决定程序的流向。

# 1.3 微处理器组成



## 1.3.3 内部寄存器

### 5. 地址寄存器 AR (Address Register)

存放正要取出的指令的地址或操作数的地址，根据此地址从存储器中取出指令。

**取指令时：**将PC中存放的指令地址送至AR，根据此地址从M中取出指令；

**取操作数时：**将操作数地址通过内部DB送到AR，再根据此地址从M中取出操作数；

**向存储器存入数据时：**首先将待写入数据的地址送到AR，再根据此地址向M写入数据。

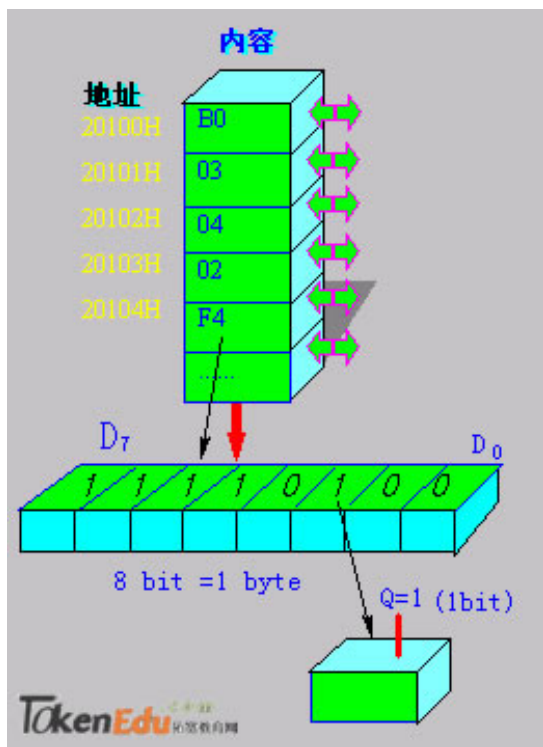
### 6. 寄存器阵列 RA (Register Array)

若干通用寄存器和专用寄存器。其设置和微处理器型号有关。



# 1.4 存储器概述

## 1.4.1 基本概念



存储器是微机的**存储和记忆部件**，用来存放程序和数据（包括原始数据、中间结果与最终结果）。

在计算机内部，程序和数据都是用0、1**二进制代码的形式**来表示的。每一个0或1就叫做1位信息。



# 1.4 存储器概述

## 1.4.1 基本概念

**字节(byte):** 8位二进制代码为1个字节。

**字(word):** 2个字节组成1个字，表示16位数据。

**字长:** 计量处理二进制代码位长的单位；  
表示计算机数据总线上1次能够处理的信息的位数；  
如：8位机，16位机，32位机等。

**内存(主存):** 内存划分为很多个存储单元(称内存单元)。  
每个存储单元存放1个字节的二进制信息；  
表示数据或指令。

**存储容量:** 存储单元的总数目，取决于地址线的根数。

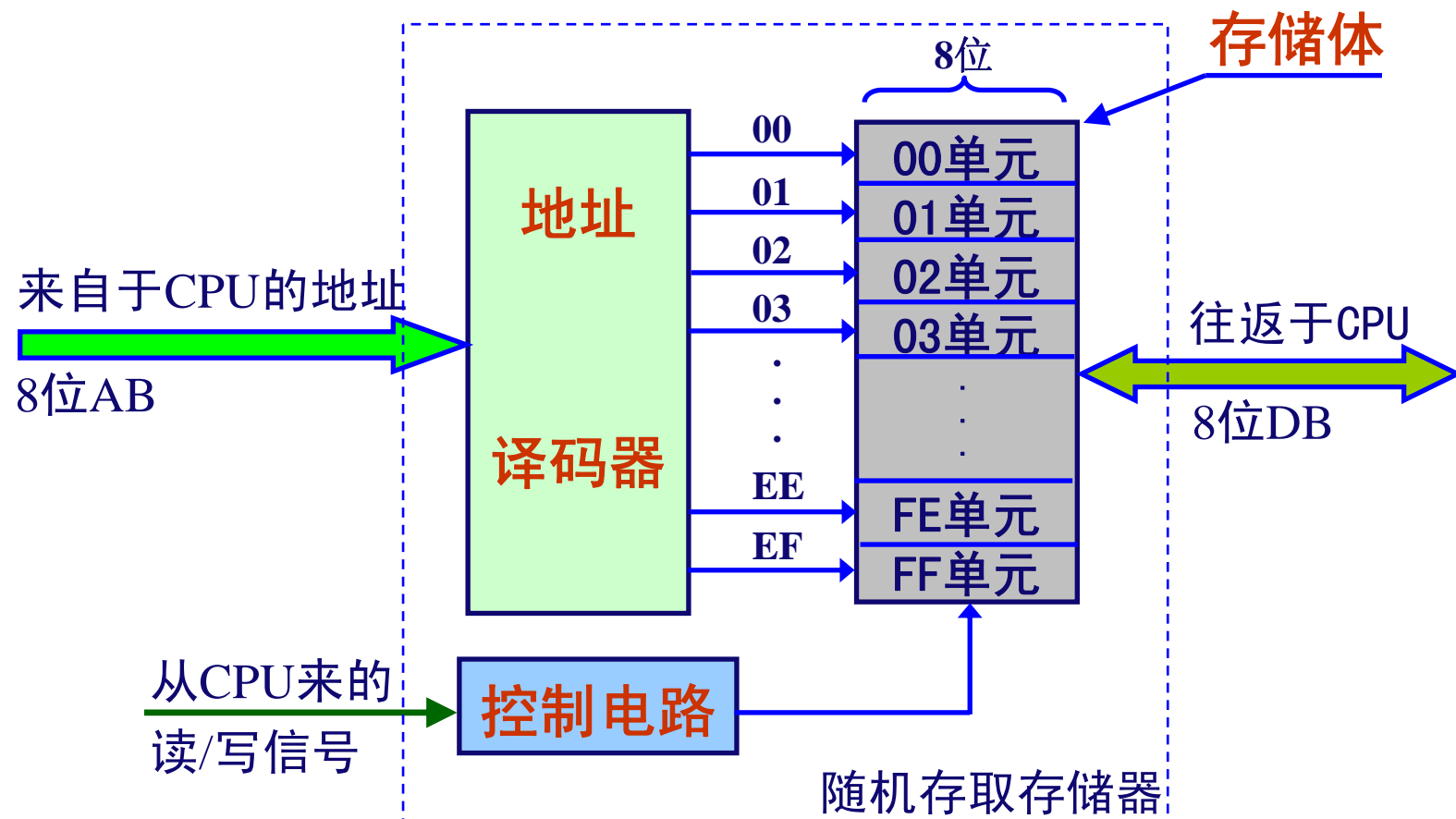
**地址:** 赋予每个存储单元1个编号，通常用16进制数表示。



# 1.4 存储器概述

## 1.4.2 存储器组成

假定存储器由256个单元组成，每个单元存储8位二进制信息，即256×8位的读/写存储器。





# 1.4 存储器概述

## 1.4.2 存储器组成

随机存取存储器由**存储体**、**地址译码器**、**控制电路**组成。

**随机存取：**所有存储单元均可随机被访问，既可以读出也可以写入信息。

**存储体：**若有**8**根地址线，存储容量为 **$2^8=256$** 个单元，  
地址编号00H~FFH（0000 0000~1111 1111）

**地址译码器：**接收来自AB送来的地址码，  
经译码器译码，选中相应的某个存储单元，  
从中读出信息或写入信息。

**控制电路：**控制存储器的读/写操作过程。

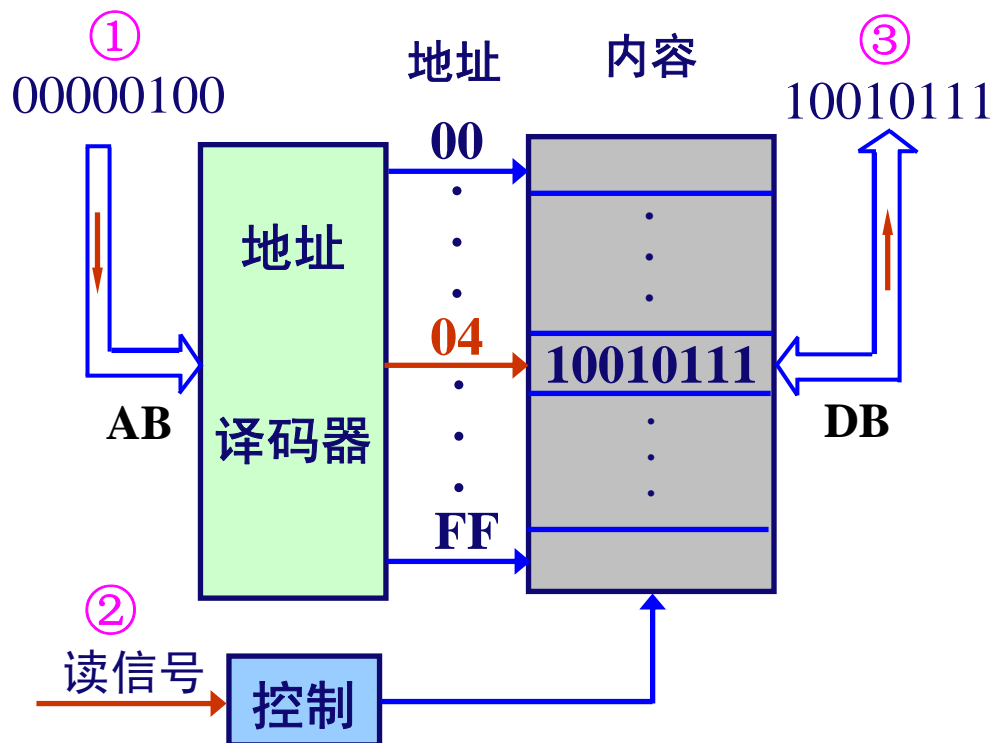


# 1.4 存储器概述

## 1.4.3 读操作过程 (非破坏性读出 non destructive read out, NDRO)

- ① CPU的AR先将地址04H放到AB上, 经地址译码器译码选中04H单元;
- ② CPU发出“读”信号给M, 指示它准备把被寻址的04H单元中的内容97H放到DB上;
- ③ 在“读”信号作用下, M将04H单元内容97H放到DB上, 经它送至DR, 然后由CPU取走该内容。

例如: CPU要读出M的04H单元  
内容10010111 (97H)。







# 1.4 存储器概述

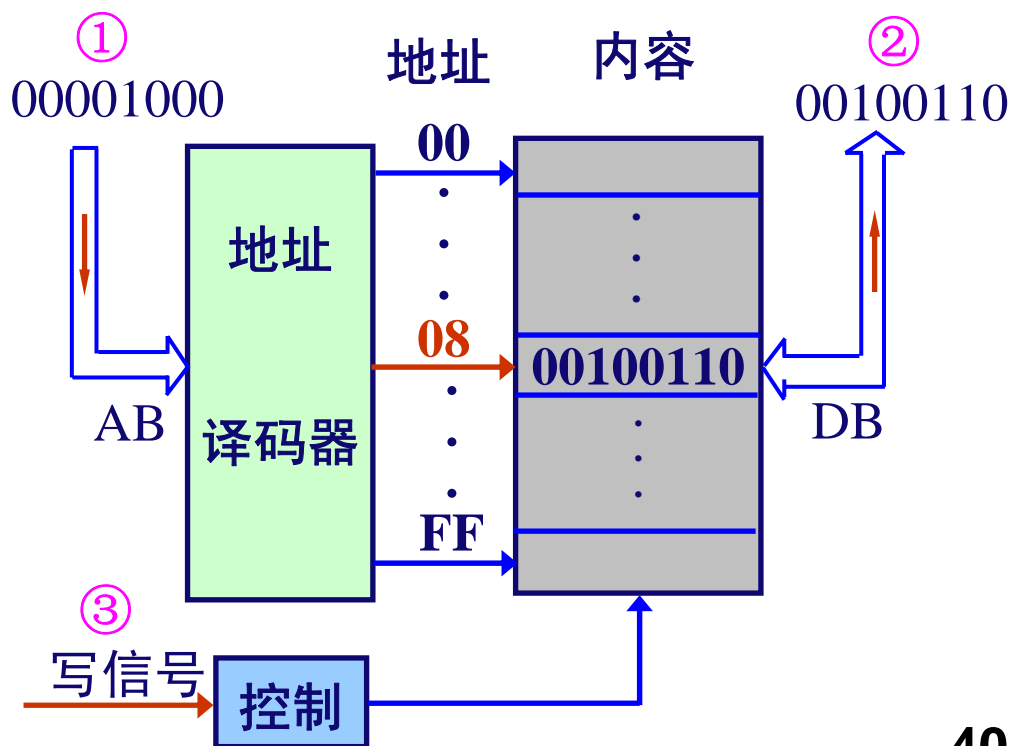
## 1.4.3 写操作过程 (破坏单元中原来的内容)

① CPU的AR先将地址08H放到AB上, 经地址译码器译码选中08H单元;

例如: CPU要把DR中的内容00100110 (26H), 写入M的08H单元。

② CPU把DR中的内容26H放到DB上;

③ CPU发出“写”信号给M, 将内容26H写入被寻址的08H单元。



# 1.5 微机工作过程



微机的工作过程就是执行程序的过程，而程序由指令序列组成。

执行程序的过程，就是执行指令序列的过程。

执行每一条指令，都包括取指令与执行指令两个基本阶段，因此微机的工作过程，也就是不断地取指令和执行指令的过程。



The diagram illustrates the instruction cycle. On the left, a vertical flow shows the program execution: **开始** (Start) in an orange box, followed by **程序** (Program) in a light blue box, and **结束** (End) in a purple box. Arrows from the **开始** and **程序** boxes point to a large grey box representing memory. Inside this memory box, a sequence of instructions is shown: **指令1** (Instruction 1), **指令2** (Instruction 2), vertical dots, and **指令n** (Instruction n). Above the memory box, red text reads: 存储器中存储的一系列指令 (A series of instructions stored in memory). To the right of the memory box, a callout box shows the **指令格式** (Instruction Format) with **操作码** (Opcode) in a pink box and **操作数** (Operand) in a purple box, with the note (地址码) (Address code) below. Arrows from **指令1** and **指令n** point to a dashed blue box on the right labeled **指令周期** (Instruction Cycle). Inside this box, the stages of the cycle are shown for each instruction. For **指令1**, the stages are **取出指令** (Fetch instruction) in a green box and **指令译码** (Instruction decoding) in a light green box, grouped by a bracket labeled **取指** (Fetch). For **指令n**, the stages are **操作数读出** (Operand reading) in a yellow box and **指令执行** (Instruction execution) in a yellow box, grouped by a bracket labeled **执指** (Execute). The **操作数读出** stage is also labeled with (地址码) (Address code) below it.

# 1.5 微机工作过程



在取指阶段，CPU从内存中读出的内容必为指令。于是，DR便把它送至IR；然后由指令译码器译码，控制器发出相应的控制信号，CPU便知道该条指令要执行什么操作。

在取指阶段结束后，机器就进入执指阶段，这时，CPU执行指令所规定的具体操作。当一条指令执行完毕以后，就转入了下一条指令的取指阶段。

周而复始地循环，一直进行到程序中遇到暂停指令时才结束。

# 1.5 微机工作过程



**取指阶段**是由一系列相同的操作组成的, 因此取指阶段的时间总是相同的, 它称为**公操作**。

**执指阶段**由不同的事件顺序组成, 取决于被执行指令的类型, 因此执指阶段的时间从一条指令到下一条指令变化相当大。

指令通常包括**操作码**(Operation CODE)和**操作数**(OPerand)。

**操作码**: 表示计算机执行什么具体操作。

**操作数**: 表示参加操作的数的本身或操作数所在的地址, 也称之为**地址码**。

8位机中, 1个存储单元只能存放1个字节, 而指令根据其所含内容不同而有**单**字节、**双**字节、**3**字节、最多**4**字节之分。

# 程序执行过程实例



例：计算 $3+2=?$

编写程序前, 须首先查阅所使用的微处理器的**指令表**, 它是某种微处理器所能执行的全部操作命令汇总。**不同系列的微处理器各自具有不同的指令表。**

人们给每条指令规定了一个缩写词, 称作**助记符**。

**机器码**用二进制和十六进制两种形式表示, 计算机和程序员用它来表示指令。

TokenEdu 拓客教育网				
表1.1 模型机指令表之一				
名 称	助 记 符	机 器 码		说 明
立即数取入累加器	MOV A , n	10110000 n	B0 n	这是一条双字节指令, 把指令第2字节的立即数n取入累加器A中。
加立即数	ADD A , n	00000100 n	04 n	这是一条双字节指令, 把指令第2字节的立即数n与A中的内容相加, 结果暂存A。
暂 停	HLT	11110100	F4	停止所有操作

# 程序执行过程实例



根据指令表，用助记符形式和十进制数表示的加法运算的

程序可表达为：  
MOV A, 3  
ADD A, 2  
HLT

模型机并不认识助记符和十进制数，只认识用二进制数表示的操作码和操作数。因此，必须按二进制数的形式来写程序，即用对应的操作码代替每个助记符，用相应的二进制数代替每个十进制数。

MOV A,3 变成	1011 0000 0000 0011	操作码(MOV A ,n) 操作数(3)
ADD A,2 变成	0000 0100 0000 0010	操作码(ADD A , n) 操作数(2)
HLT 变成	1111 0100	操作码(HLT)



# 程序执行过程实例



整个程序是3条指令5个字节。

微处理器和存储器均用8位字或1个字节存放与处理信息，把这段程序存入存储器时，共需要占5个存储单元。

假设把它存放在存储器最前面的5个单元里，则该程序将占有从00H~04H这5个单元。

地址		指令的内容	助记符内容
十六进制	二进制		
00	0000 0000	1011 0000	MOV A, n
01	0000 0001	0000 0011	03
02	0000 0010	0000 0100	ADD A, n
03	0000 0011	0000 0010	02
04	0000 0100	1111 0100	HLT
.	.		
.	.	.	
.	.	.	
FF	1111 1111		

# 程序执行过程实例

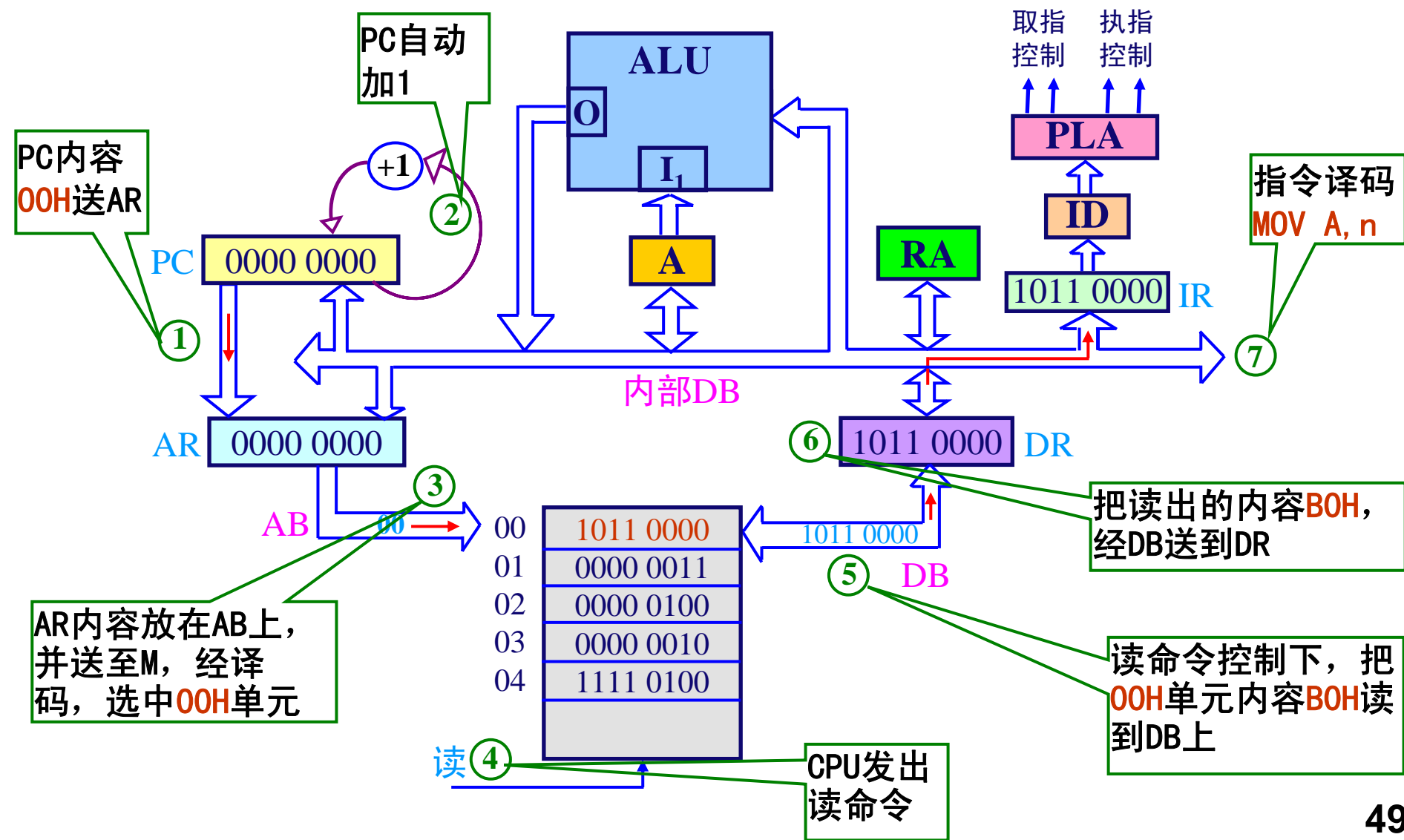


当程序存入存储器后，开始微机内部执行程序的具体操作过程。

开始执行程序时，必须先给程序计数器PC赋以第1条指令的首地址00H，然后就进入第1条指令的取指阶段。

取第1条指令的第1字节的操作过程如下：

# 程序执行过程实例 (取第1条指令的操作)



# 程序执行过程实例



- ① 把PC的内容00H送到AR。
- ② 一旦PC的内容可靠地送入AR后，PC自动加1，即由00H变为01H。注意，此时AR的内容并没有变化。
- ③ 把AR的内容00H放在AB上，并送至M，经地址译码器译码，选中相应的00H单元。
- ④ CPU发出读命令。

# 程序执行过程实例



- ⑤ 在读命令控制下，把所选中的00H单元中内容即第1条指令的操作码B0H 读到DB上。
- ⑥ 把读出的内容B0H经DB送到DR。
- ⑦ 取指阶段的最后一步是指令译码。因为取出的是指令的操作码，故DR把它送到IR,然后再送到ID。

经过译码，CPU“识别”出这个操作码B0H就是MOV A,n指令，于是，它“通知”控制器发出执行这条指令的各种控制命令。这就完成了第1条指令的取指阶段。

然后转入执行第1条指令的阶段。

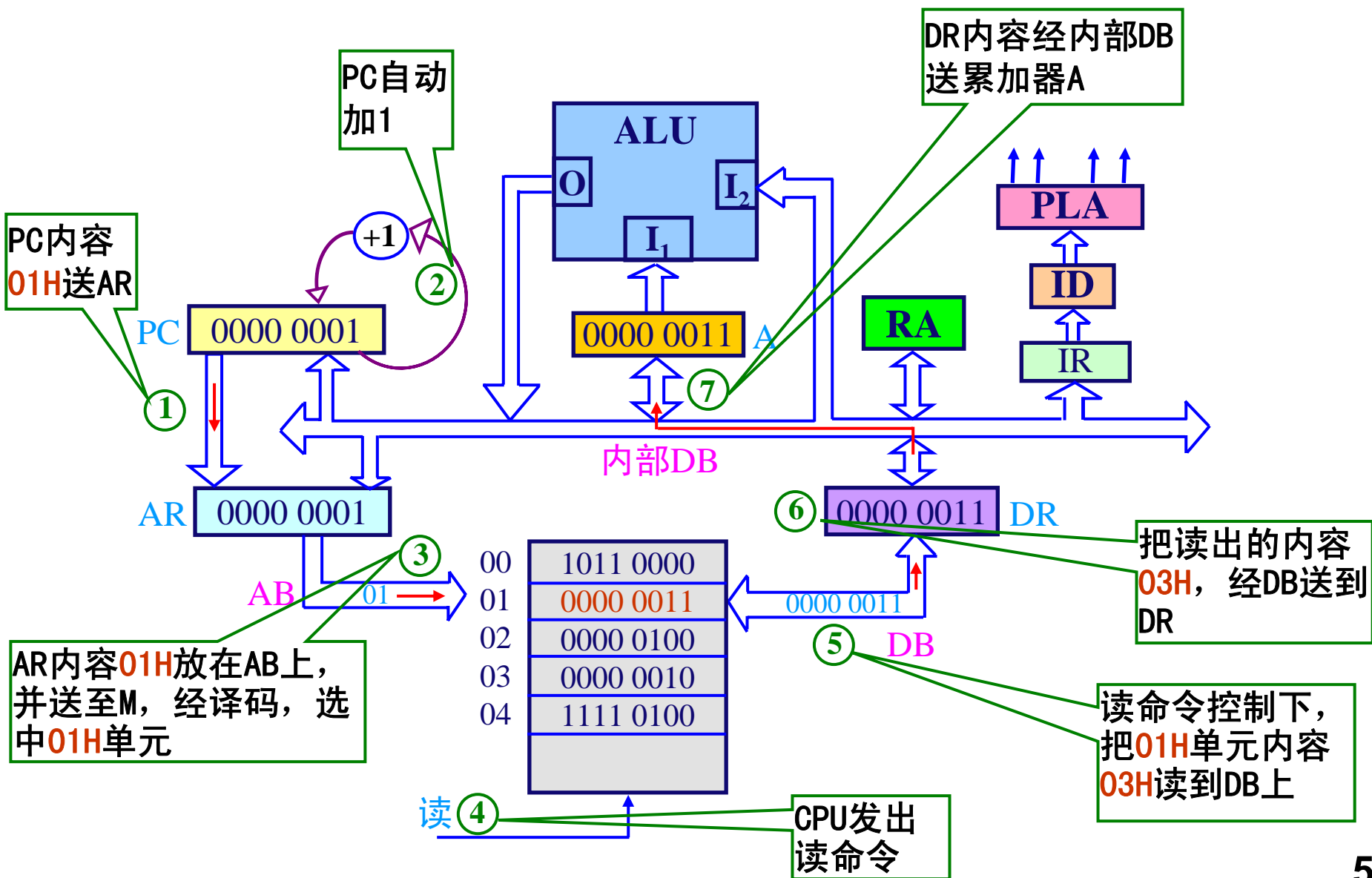
# 程序执行过程实例



经过对操作码B0H译码后，CPU 就“知道”这是一条把下一单元中的操作数取入累加器A的双字节指令MOV A,n，所以执行第1条指令就必须把指令第2字节中的操作数03H取出来。

取第1条指令的第2字节的过程如下：

# 程序执行过程实例 (取立即数的操作)



# 程序执行过程实例



- ① 把PC的内容01H送到AR。
- ② 当PC的内容可靠地送到AR后，PC自动加1，变为02H。  
但这时AR中的内容01H并未变化。
- ③ AR通过AB把地址01H送到存储器的地址译码器，经过译码选中相应的01H单元。
- ④ CPU发出读命令。



# 程序执行过程实例



- ⑤ 在读命令控制下，将选中的01H单元的内容03H读到DB上。
- ⑥ 通过DB把读出的内容送到DR。
- ⑦ CPU根据该条指令具有的字节数已知此时读出的是操作数，且指令要求把它送到累加器A，故由DR取出的内容就通过内部DB送到累加器A。

于是第1次执指阶段完毕，操作数03H被取入累加器A中；并进入第2条指令的取指阶段。

# 程序执行过程实例

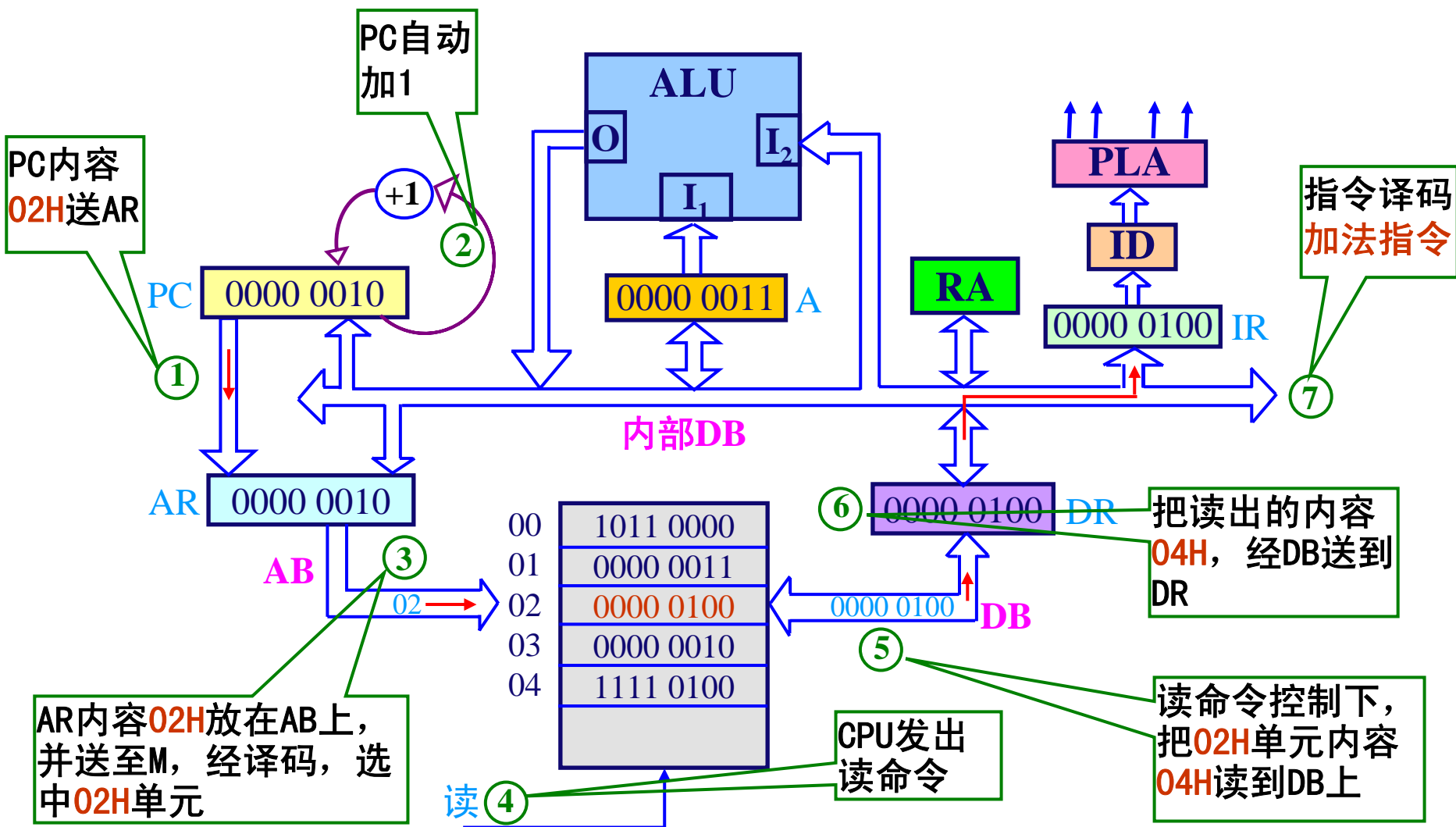


取第2条指令的过程与取第1条指令的过程相同，只是在取指阶段的最后一步，读出的指令操作码04H由DR把它送到IR，经过译码发出相应的控制信息。

当ID对指令译码后，CPU就“知道”操作码04H表示一条加法指令，意即以累加器 A 中的内容作为一个操作数，另一个操作数在指令的第2字节中。

取第2条指令的第1字节的过程如下：

# 程序执行过程实例 (取第2条指令的操作)



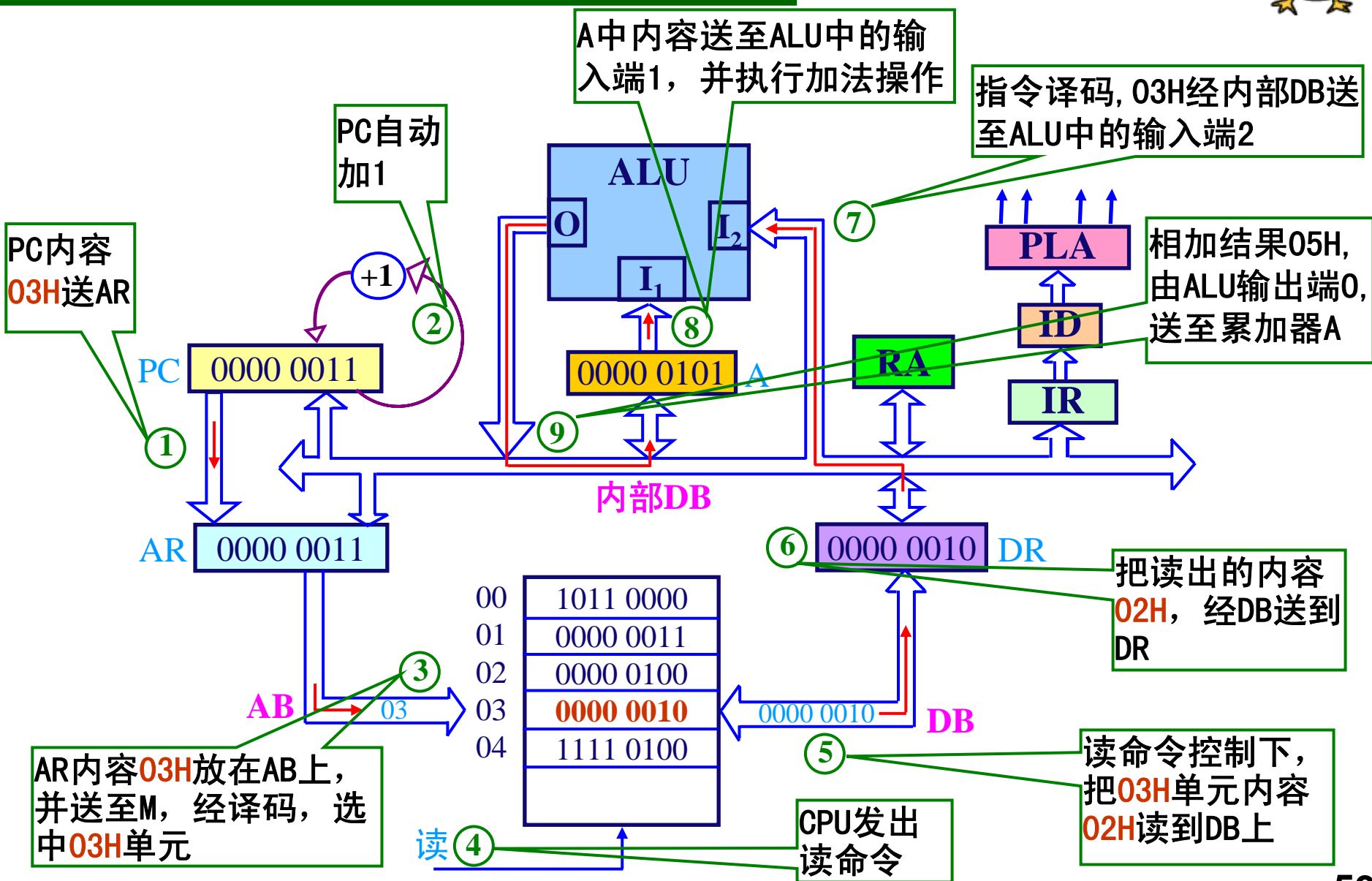
# 程序执行过程实例



执行第2条指令，必须取出指令的第2字节。

取第2条指令的第2个字节及执行指令的过程如下：

# 程序执行过程实例 (执行第2条指令的操作)



# 程序执行过程实例



- ① 把PC的内容03H送到AR。
- ② 当把PC的内容可靠地送到AR后，PC自动加1。
- ③ AR通过AB把地址号03H送到ID，经过译码，选中相应的03H单元。
- ④ CPU发出读命令。
- ⑤ 在读命令控制下，把选中的03H单元中的内容即数02H读至DB上。
- ⑥ 数据通过DB送到DR。

# 程序执行过程实例



- ⑦ 因在对指令**译码**时, **CPU**已知读出的数据02H为操作数, 且要将它与已暂存于A中的内容03H相加, 故数据由DR通过内部DB送至ALU的另一输入端**I<sub>2</sub>**。
- ⑧ A中的内容送ALU的输入端**I<sub>1</sub>**, 且执行**加法**操作。
- ⑨ 把相加的结果05H由ALU的输出端**O**又送到累加器**A**中。

至此,第2条指令的执行阶段结束, A中存入和数5H,而将原有内容03H冲掉。接着,就转入第3条指令的取指阶段。

# 程序执行过程实例



程序中的最后一条指令是HLT。可用类似上面的取指过程把它取出。

当把HLT指令的操作码F4H取入DR后,因是取指阶段,故CPU将操作码F4H送IR, 再送指令译码器ID; 经译码, CPU“已知”是暂停指令, 于是控制器停止产生各种控制命令, 使计算机停止全部操作。

此时,程序已完成 $3+2$ 的运算, 并且和数5已放在累加器中。



# 1.6 微机系统的主要性能指标



**主要选项：**主板+CPU、硬盘、显示器、显卡、声卡与光驱等多媒体套件、内存以及机箱等。

**微机系统性能：**主板与CPU、外设配置、总线结构以及软件配置

微机系统的主要性能指标分以下几个方面：

- ① 最佳整体性能评估与最佳购买性能评估
- ② 主板的结构与性能
- ③ 微处理器的性能指标
- ④ 硬盘的性能指标
- ⑤ 允许配置的外设数量
- ⑥ 总线的性能指标
- ⑦ 软件的配置

# 1.6 微机系统的主要性能指标



## 1. 最佳整体性能评估与最佳购买性能评估

### ◆ 最佳整体性能评估 (5方面)

性能(67%)、可用性(17%)、特征(8%)、技术创新(4%)、  
价格(4%)

### ◆ 最佳购买性能评估

与整体性能评估相比，其评估项目基本相同，但各项权重不同。  
价格(50%)、性能(30%)、可用性(10%)、特征(10%)

# 1.6 微机系统的主要性能指标



## 2. 主板的结构与性能

### ◆ 主板(mother board)

主板又称为母板、主机板、系统板等，是微机硬件系统中最重要部件，其结构与性能如何将直接影响到微机各个部件之间的相互配合及其整体性能。

**主板上集中了微机的主要部件和接口电路：**

CPU、内存条和高速缓存芯片、系统芯片组等**直接**装在主板上；  
硬盘、软驱和光驱通过**数据总线**与主板相连；  
鼠标、键盘和各种扩充卡经**外设接口或扩展槽**装或接插在主板上。

# 1.6 微机系统的主要性能指标

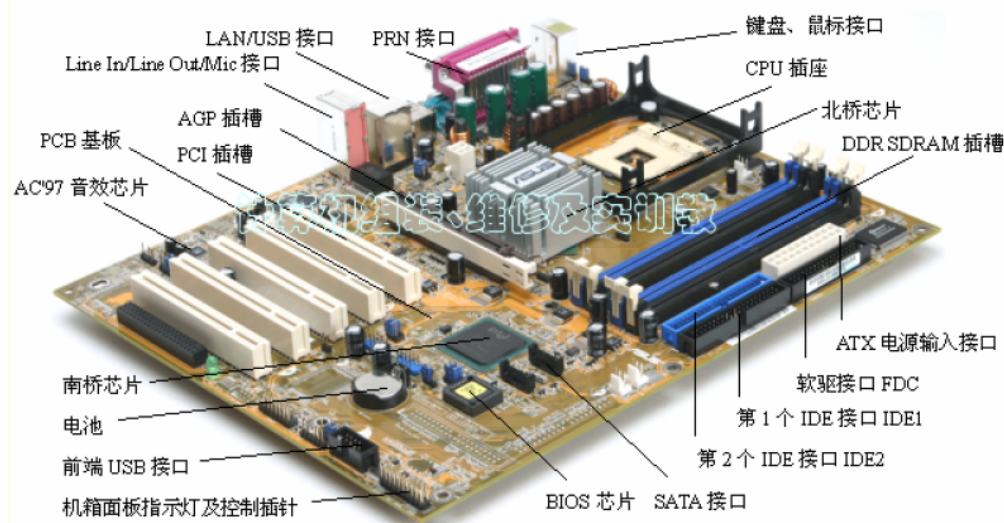


## 2. 主板的结构与性能

### ◆ 主板结构

主板结构：根据主板上各元器件的布局排列方式，尺寸大小，形状，所使用的电源规格等制定出的通用标准，所有主板厂商都必须遵循。

现在主板结构主要有AT、ATX (AT extended) 两种基本类型，NLX是最新结构的主板。





# 1.6 微机系统的主要性能指标



## 2. 主板的结构与性能



Micro-ATX主板俗称“小板”，它在结构上与标准ATX主板相同，区别只是Micro-ATX比标准ATX主板减少了部分扩展插槽，这样尺寸也就减小了不少，降低了厂家的生产成本和用户组装电脑时的投资。



标准ATX主板的尺寸通常“横长竖短”，俗称“大板”，主要特点是将串口、并口、鼠标和键盘接口都固定在主板上，内置声卡功能时还将声卡的接口也一并做在主板上。另外ATX主板必须使用ATX结构的机箱电源，这样才能保证ATX主板的定时开机、Modem唤醒、键盘开机等特殊功能的实现。

# 1.6 微机系统的主要性能指标



## 2. 主板的结构与性能

**NLX 结构**是IBM 公司与Intel 公司共同开发的**主板结构标准**，是新一代一体化主板结构规范。NLX 是一种灵活的规范，它通过定义基本形状，比如尺寸和安装方式等来帮助确保其兼容性，但给计算机制者留下了自由发挥的空间。

NLX 结构的**最大特点**在于其Add-in 卡，它直接固定在机箱上，上面有PCI 和ISA 插槽，以及软驱和IDE 接口，为主板供电的电源接口也在它的前端，而**主板则像一块附加卡一样插到Add-in 卡上**，安装和更换都很方便。

由于主板集成了连接各主要外部设备的接口，基本上可以不再使用接口插卡，提高了系统集成度和稳定性。

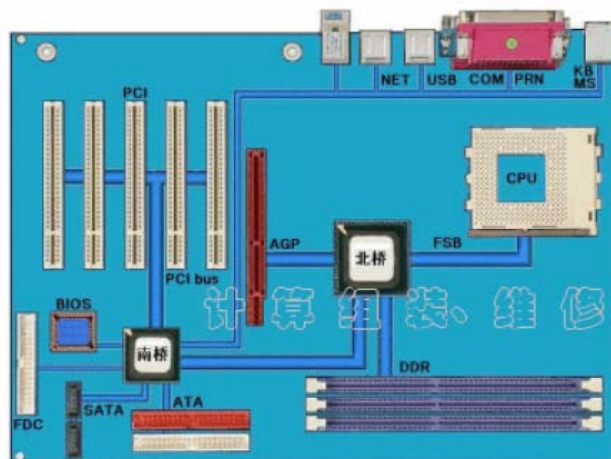
# 1.6 微机系统的主要性能指标



## 2. 主板的结构与性能

### ◆ 生产主板芯片组 (Chipset) 的厂家

中央处理器(CPU)是整个电脑系统的核心，芯片组是整个身体的躯干。目前为止，能够生产芯片组的厂家有英特尔(美国)、VIA(威盛,中国台湾)、SiS(矽统,中国台湾)、ULI(中国台湾)、AMD(美国)、NVIDIA(美国)、ATI(加拿大)、ServerWorks(美国)、IBM(美国)、HP(美国)等，其中以**英特尔和NVIDIA及VIA的芯片组**最为常见。



# 1.6 微机系统的主要性能指标



## 2. 主板的结构与性能（续）

### ◆ 生产主板的厂家

微星(MSI)、华硕(ASUS)、佰钰(Acorp)、建基(Aopen)、磐正(EPoX)、升技(Abit)、硕泰克(Soltek)、映泰(CIOSTAR)、捷波(Jetway)、技嘉(Gigabyte)、联想(QDI)等。



# 1.6 微机系统的主要性能指标



## 2. 主板的结构与性能

### ◆ CPU在主板上面的安装结构形式

CPU需经某个接口与主板连接才能进行工作，接口方式有引脚式、卡式、触点式、针脚式等。

目前CPU的接口都是针脚式接口，对应到主板上就有相应的插槽类型。分为Socket和Slot两大类。

Socket结构类的主板可支持的CPU：Intel的Pentium、新版Celeron(赛扬)、AMD的K5/K6系列、Cyrix的6x86系列等。其成本比Slot低。

Slot机构最先是由Intel公司为配合Pentium II 和部分Celeron微处理器而推出的CPU接插结构，分为Slot 1和Slot 2。

# 1.6 微机系统的主要性能指标



## 2. 主板的结构与性能

### ◆ 主板的主要性能指标

- ① 微处理器支持的**能力**：包括CPU插槽类型、CPU种类、外频范围、电压范围；
- ② 系统芯片组的**类型**；
- ③ 是否**集成**显卡、声卡、调制解调器、网卡；
- ④ 支持内存和高速缓存的**类型与容量**；
- ⑤ 系统BIOS的版本、功能，**是否支持**即插即用；
- ⑥ 扩充插槽及I/O接口的**数量、类型**；
- ⑦ 主板的**电压**输出范围。

# 1.6 微机系统的主要性能指标



## 3. 微处理器的性能指标

◆ **字长：** CPU内部一次能同时处理的数据的**位数**。

字长标志计算精度，字长越长，精度越高，但制造工艺越复杂。

微机的字长有1、4、8、16、32位等多种，相应的就有1位机、4位机、8位机、16位机以及32位机等。目前，一般用户最常用的是32位机。

◆ **内存容量：** 以字节为单位计算的。

8位机中，16条地址线，寻址范围是 $2^{16}=64\text{KB}$ 。

16位机中，20条地址线，寻址范围是 $2^{20}=1024\text{KB}=1\text{MB}$ 。

32位机中，32条地址线，寻址范围是 $2^{32}=4\text{GB}=4*1024\text{MB}$

# 1.6 微机系统的主要性能指标



## 3. 微处理器的性能指标

◆ **指令系统：**微机的核心部件微处理器都有各自的指令系统。

16位微处理器Intel 8086，有24种寻址方式，148条基本指令，能满足单任务、单用户系统的多种应用需要。

1985年和1989年，Intel 先后推出了32位微处理器 80386与80486，能胜任多任务与多用户系统的复杂任务。

1993年至今，Intel连续推出了Pentium/Pentium MMX/PRO/II/III/4系列微处理器，能顺利实现对多媒体信息的实时处理。

# 1.6 微机系统的主要性能指标



## 3. 微处理器的性能指标

◆ **运算速度**：微机性能综合体现，微处理器执行指令的速率。

运算速度用以衡量计算机运算的快慢程度，与诸多因素有关，如机器主频、执行何种操作、主存本身速度等。

- **平均速度**：根据不同类型指令在计算过程中出现的频繁程度，乘上不同系数，求得统计平均值。
- 以**执行时间最短的指令**的标准来计算速度。
- 直接给出每条指令的**实际执行时间**和**机器的主频**。

# 1.6 微机系统的主要性能指标



## 3. 微处理器的性能指标

◆ **主频**：指 CPU 的时钟频率，以Hz为单位。是判定微处理器执行性能的一个基本指标。

8086 最高主频为 10MHz

Pentium 主频为 100MHz

Pentium II 最高主频为 450MHz

Pentium III 最高主频为 850MHz

Pentium 4 最高主频已达到 3.8GHz

**MIPS** (Million Instruction Per Second)：每秒内执行的百万条指令数，是微处理器执行速度的一种量度方式。

8086执行速度为 0.4~1.3 MIPS

Pentium III 执行速度为 300 MIPS

# 1.6 微机系统的主要性能指标



## 3. 微处理器的性能指标

◆ **iCOMP**: intel Comparative Microprocessor Performance index  
是衡量Intel系列微处理器性能的比较指数。

16位**整数**运算占 53%;

16位**浮点**运算占 2%;

32位**整数**运算占 15%;

16位**图形处理**占 10%;

32位**浮点运算与图形处理**各占 5%;

16位及32位**图像处理**各占 5%。

**iCOMP**值以工作频率为25MHz的80486SX-25为基准，将其指数定位100，其它微处理器与之相比，计算出相对值。

# 1.6 微机系统的主要性能指标



## 3. 微处理器的性能指标

◆ **SPECmark**: 是 SPECint (系统性能**整数数值**计算法) 与 SPECfp (系统性能**浮点数值**计算法) 的合称。

它们分别由多项**实际应用基准测试**组成, 所涉及领域包括: 电路理论、逻辑设计、文字压缩算法、电子表格、软件开发与LISP编译器等。



# 1.6 微机系统的主要性能指标



## 4. 硬盘的性能指标

**容量：**用户优先考虑的指标。

以MB (兆)和GB(千兆)为单位，硬盘的容量有40GB、60GB、80GB、100GB、120GB、160GB、200GB等。

**速度：**硬盘速度在微机系统中的作用仅次于CPU和内存。

**主轴电机转速：**4200/5400/7200/10000 rpm

**硬盘缓存容量：**2MB和8MB缓存是现今主流硬盘所采用，而在服务器或特殊应用领域中缓存容量甚至达到了16MB、64MB等。

**安全性：**提高抗外界震动或抗瞬间冲击以及数据传输纠错。  
众多厂家开发了一些硬盘安全技术和软件。

# 1.6 微机系统的主要性能指标



## 5. 允许配置的外设数量

允许挂接的外设数量越多，微机的功能越强。

- 常规的人机交互设备：如键盘、鼠标、显示器、打印机等。
- 常用通讯设备：如网卡和调制解调器。
- 扫描仪

# 1.6 微机系统的主要性能指标



## 6. 总线的性能指标

总线结构是微机性能的重要指标之一。

- ① **ISA** (Industry Standard Architecture) 是**工业标准体系结构总线**的简称，是PC/AT机及其兼容机所使用的**16位**标准系统扩展总线，又称**PC-AT**总线，其数据传输率为16MB/s。
- ② **EISA** (Extended ISA)，数据和地址总线均增加为**32位**，数据传输率为33MB/s，适合32位微机系统。

# 1.6 微机系统的主要性能指标



## 6. 总线的性能指标

- ③ **PCI** (Peripheral Component Interconnect)是**外设互连总线**的简称，是Intel公司推出的**32/64位**标准总线，数据传输率为132MB/s，用于Pentium以上的微机系统。
- ④ **AGP** (Accelerated Graphics Port)是**加速图形端口**，是专门为提高视频带宽而设计的总线规范。数据传输率提高到266MB/s (x1模式)、532MB/s(模式)或1.064GB/s(x4模式)。

# 1.6 微机系统的主要性能指标



## 6. 总线的性能指标

总线的主要性能指标包括：

➤ 总线的带宽：

单位时间内总线上可传输的数据量，以MB/s为单位。

➤ 总线的位宽：

能同时传输的数据位数，如16位、32位、64位等。

工作频率一定的条件下，总线带宽与总线位宽成正比。

➤ 总线的工作频率：

也称为总线的时钟频率，以MHz为单位。用于协调总线上的各种操作的时钟频率。工作频率越高，则总线带宽越宽。

**总线带宽 = (总线位宽/8) x 总线工作频率 (MB/s)**

# 1.6 微机系统的主要性能指标

---



## 7. 软件的配置

指微机系统配置的操作系统和应用程序。

# Windows操作系统发展赏析

## ■ Windows 1.0

1985年11月，Microsoft Windows 1.0发布，最初售价为100美元；Microsoft Windows 1.0是微软第一次对个人电脑操作平台进行用户图形界面的尝试。Windows 1.0本质上宣告了MS-DOS操作系统的终结。



# Windows操作系统发展赏析

## ■ Windows 2.0

1987年12月9日，Windows 2.0发布。这个版本的windows图形界面，有不少地方借鉴了同期的Mac OS 中的一些设计理念，但这个版本依然没有获得用户认同。之后又推出了windows 386和windows 286版本，有所改进，并为之后的Windows 3.0的成功作好了技术铺垫。

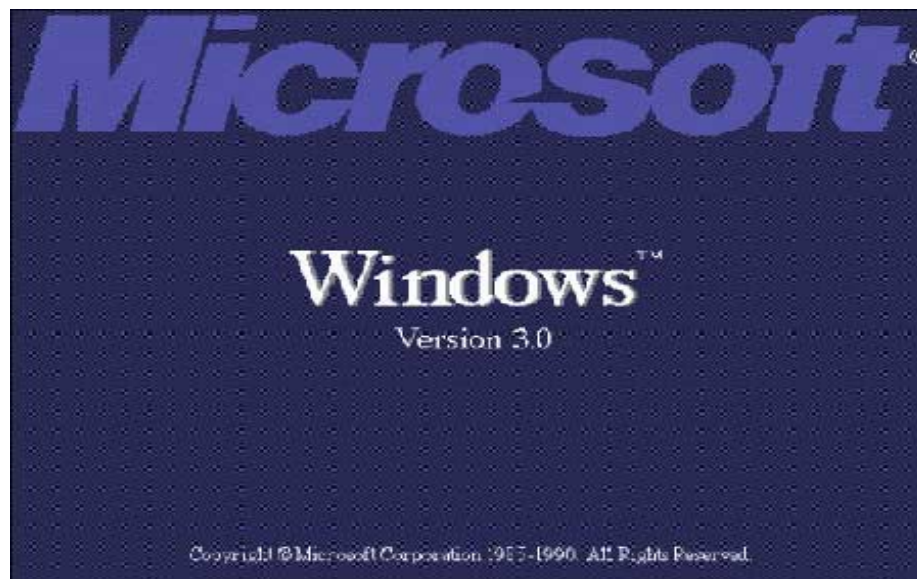




# Windows操作系统发展赏析

## ■ Windows 3.0

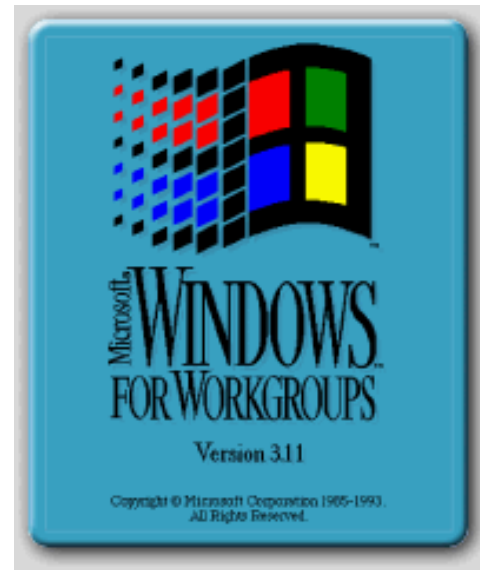
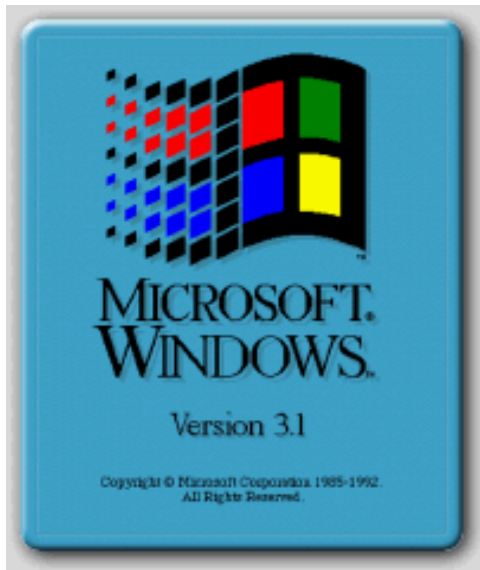
1990年5月22日，Windows 3.0正式发布，由于在界面/人性化/内存管理多方面的巨大改进，终于获得用户的认同。之后微软公司趁热打铁，于1991年10月发布了windows 3.0的多语版本，为windows在非英语母语国家的推广起到了重大作用。



# Windows操作系统发展赏析

## ■ Windows 3.1

1992年3月18日，Windows for Workgroups 3.1发布，Windows 3.1添加了对声音输入输出的基本多媒体的支持和一个CD音频播放器，以及对桌面出版很有用的TrueType字体。



# Windows操作系统发展赏析

## ■ Windows 3.2

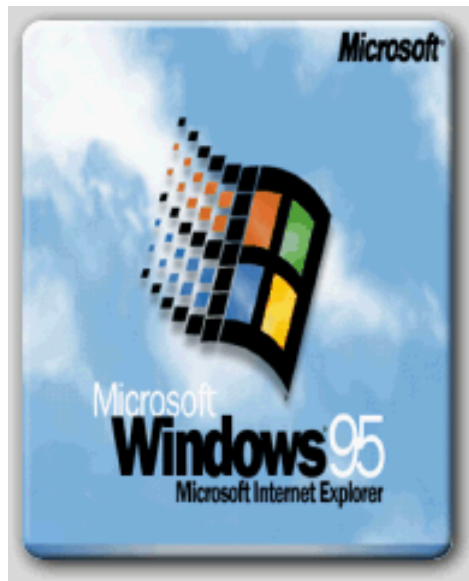
1994年，windows 3.2的中文版本发布，相信国内有不少windows 的先驱用户就是从这个版本开始接触windows系统的；由于消除了语言障碍，降低了学习门槛，因此很快在国内流行了起来。



# Windows操作系统发展赏析

## ■ Windows 95

1995年8月24日，微软推出具有里程碑意义的Windows 95。Windows 95是第一个独立的32位操作系统，并实现真正意义上的图形用户界面，使操作界面变得更加友好。



# Windows操作系统发展赏析

## ■ Windows 98

1998年6月，微软公司推出了Windows 98。与Internet的紧密集成是Windows 98最重要的特性，它使用户能够在共同的界面上以相同方式简易、快捷地访问本机硬盘、Intranet和Internet上的数据，让互联网真正走进个人应用。



# Windows操作系统发展赏析

## ■ Windows 2000

Windows 2000 Professional于2000年年初发布，是第一个基于NT技术的纯32位的Windows操作系统，实现了真正意义上的多用户。



# Windows操作系统发展赏析

## ■ Windows XP

2001年10月25日，Windows家族中极具开创性的版本Windows XP(Experience)面世。





# Windows操作系统发展赏析

---

## ■ Windows Vista

Vista是微软的下一代操作系统，以前叫做Longhorn。05年7月22日微软对外宣布正式名称是Windows Vista。这个版本的最大特色就是：贴近用户，了解用户的感受，从而方便用户。





# DOS操作系统简介

DOS是Disk Operation System (磁盘操作系统)的简称，是1985~1995年的个人电脑上使用的一种主要的操作系统。DOS操作系统首先是由IBM公司开发，称为IBM DOS；后由MicroSoft公司收购，称为MS DOS。

DOS的版本由最早的DOS 1.00版本，发展到后来的DOS 2.00、DOS 2.10、DOS 2.13、DOS 3.00、DOS 4.00、DOS 5.00、DOS 6.00、DOS 6.10、DOS 6.22 .....等等版本。

现在的WINDOWS还是建筑在它的基础上"启动"的；并保留了DOS的操作界面和使用功能。

```
D:\>dir
Volume in drive D is MS-RAMDRIVE
Directory of D:\

ATTRIB  EXE           15,252   05-13-98   17:48
CHKDSK  EXE           28,112   05-13-98   17:48
COMMAND COM          94,282   06-19-98   20:01
DEBUG   EXE          20,554   05-13-98   17:49
EDIT     COM          72,174   05-13-98   17:57
EXT      EXE          13,299   05-11-98   20:01
EXTRACT  EXE          93,242   06-19-98   20:01
FORMAT  COM           49,655   05-13-98   20:41
HELP     BAT              36   05-11-98   20:01
MSCDEX   EXE          25,473   05-11-98   20:01
README   TXT           9,868   06-19-98   20:01
RESTART  COM            20   05-11-98   20:01
SCANDISK EXE         144,211   05-14-98   10:22
SCANDISK INI           7,329   05-13-98   20:01
SYS       COM          19,159   05-13-98   20:38
        15 file(s)      592666 bytes
        0 dir(s)       1,489,920 bytes free

D:\>
```

# Question?

**第一章结束!**