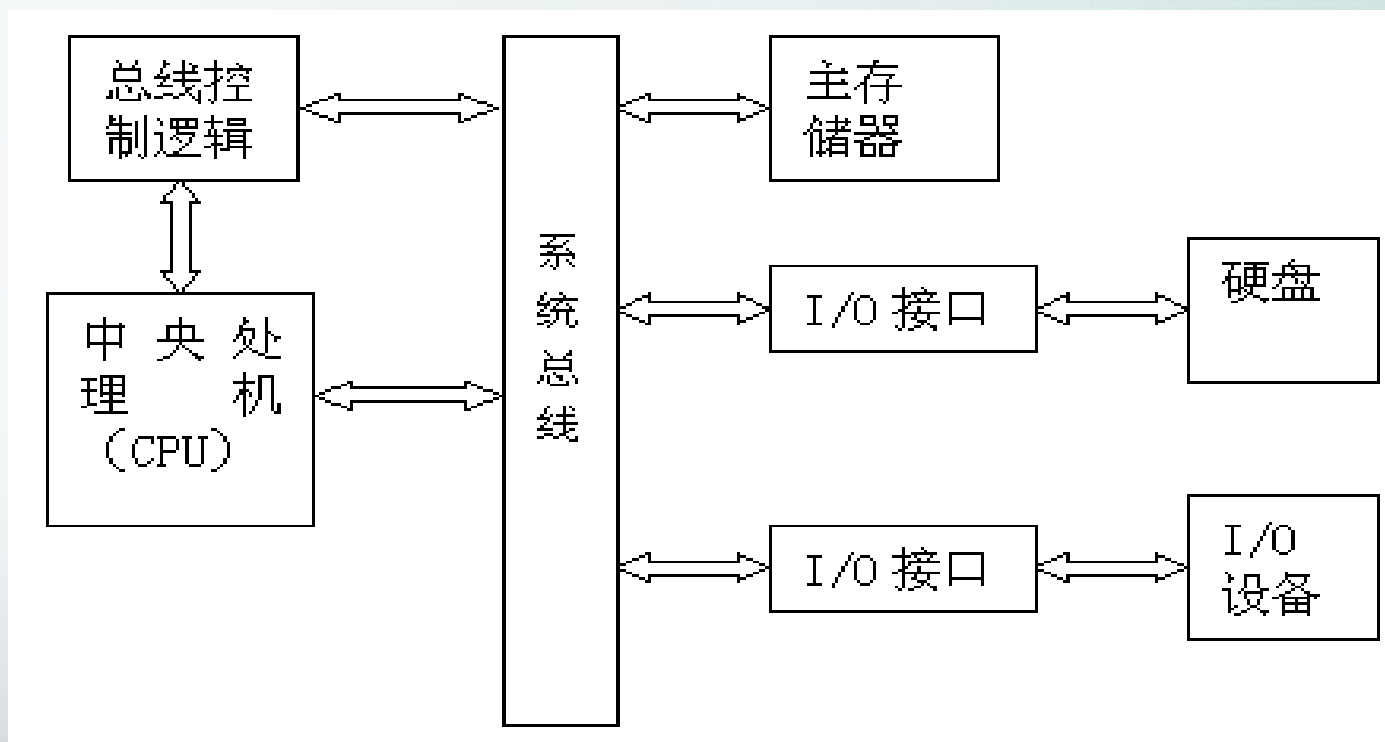


第2章 计算机基本原理

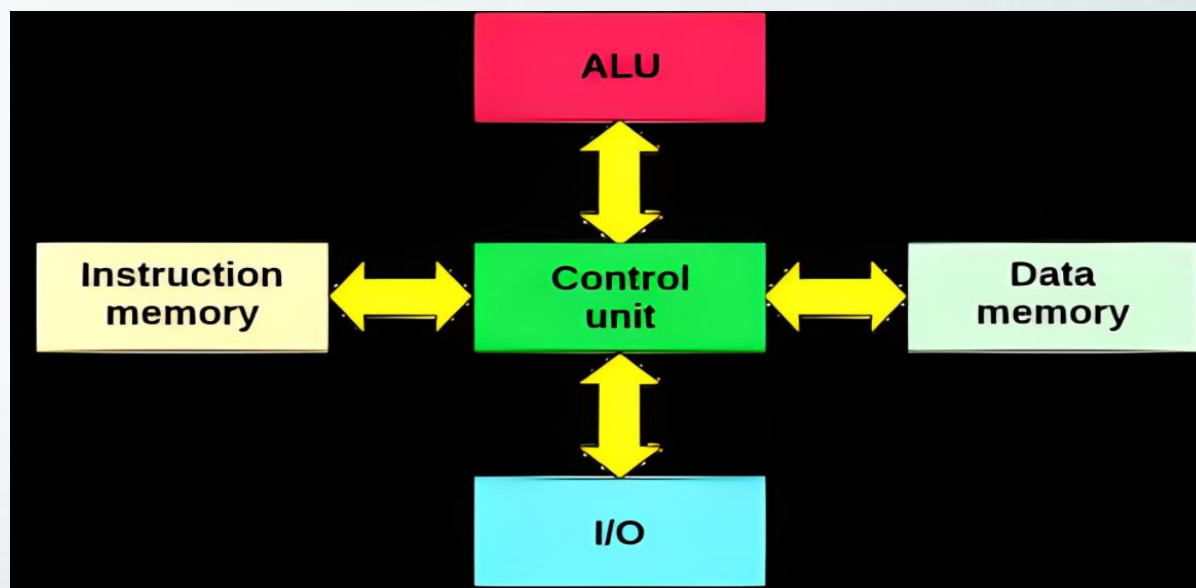
- ❖ 计算机系统组成
- ❖ 中央处理器（CPU）中的寄存器 >>
- ❖ 存储器 >>
- ❖ 外部设备和接口 >>

计算机系统组成

- ❖ 基本工作原理是存储程序和程序控制
- ❖ 冯诺依曼原理的计算机结构



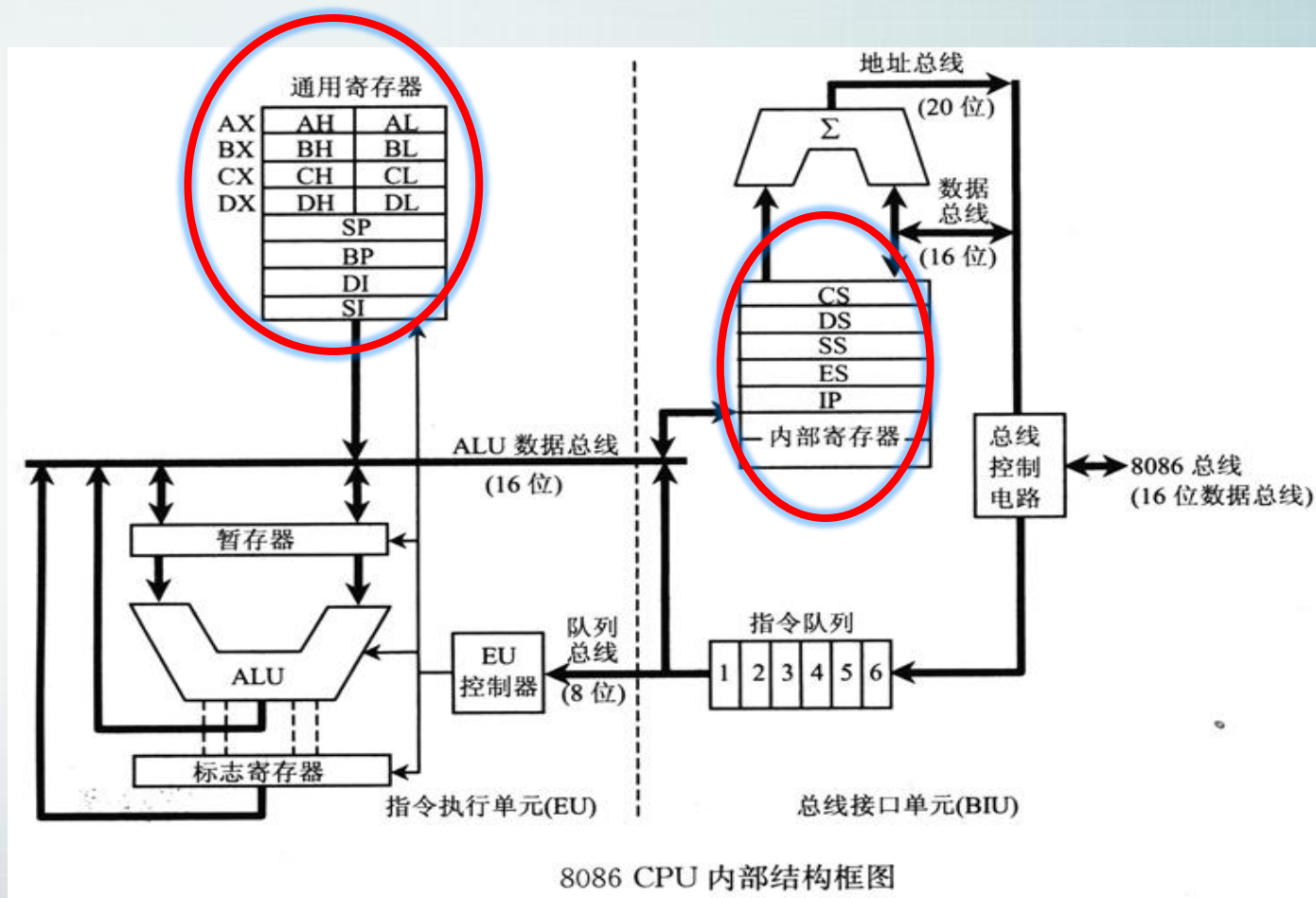
❖ 与冯诺依曼架构不一样的是，哈佛架构将数据总线和指令总线分开，降低了总线竞争的情况。

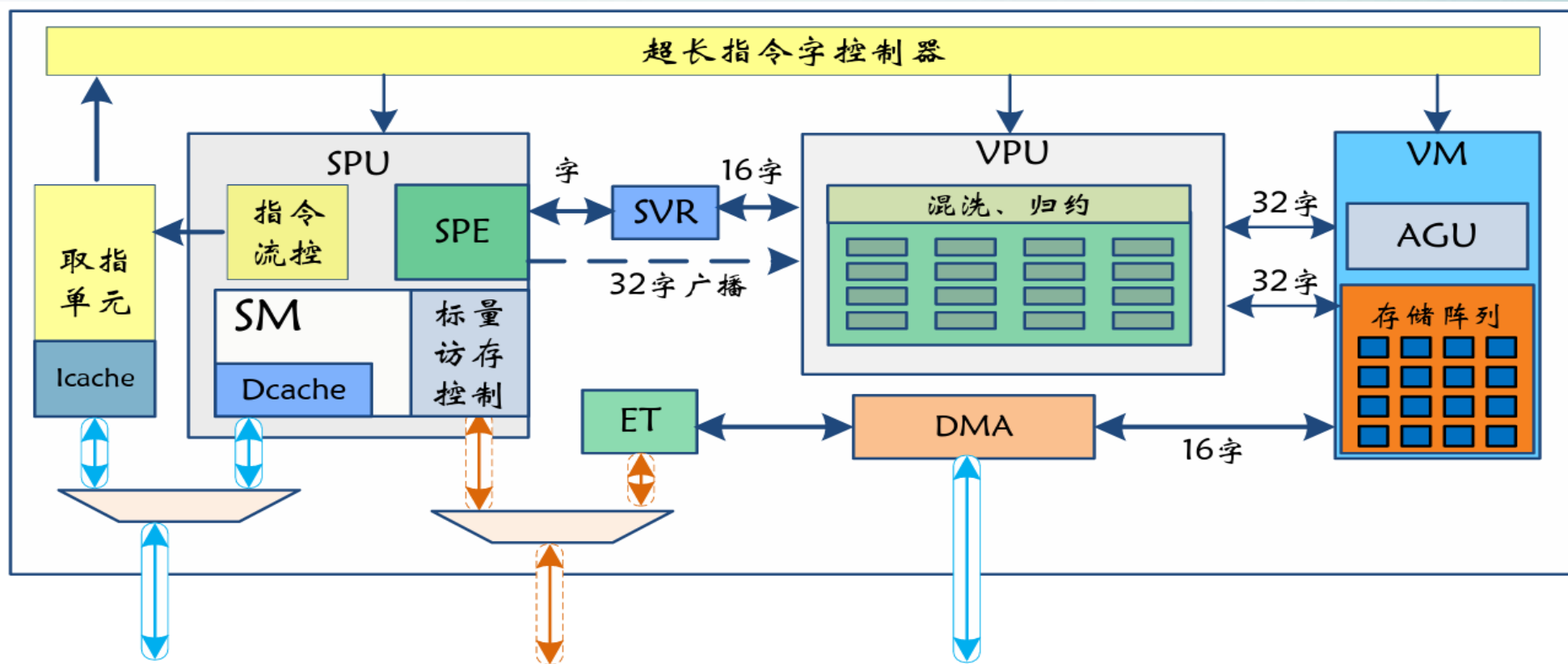



- ❖ 流水线：为了提高处理速度，流水线技术允许同时执行多个指令的不同阶段
- ❖ 超长指令字：市面上多数的DSP都是采取 VILW 指令集(超长指令字)，一次发多条指令，实现指令级的并行。
- ❖ 专用硬件单元：包含一些专用硬件单元。如当代处理器中的AI加速单元。

← CPU中的寄存器

- 一. 寄存器介绍
- 二. CS和IP
- 三. 堆栈







```
push ds
mov ax, 0
push ax
mov ax, data
mov ds, ax
mov dx, offset string
mov ah, 9
int 21h
```

16位结构的CPU

❖ 8086是16位结构的CPU。

❖ 16位结构的CPU具有以下几方面结构特征：

- ① 数据总线为16位；
- ② 运算器一次最多可以处理16位的数据；
- ③ 寄存器的最大宽度为16位；
- ④ 寄存器和运算器之间的通路为16位。



寄存器介绍

❖ 通用数据寄存器

- 8086 CPU的所有寄存器都是16位的，可以存放两个字节。
- AX、BX、CX、DX这4个寄存器通常用来存放一般性的数据，被称为通用数据寄存器，有时候也可以存放地址。

- ① **AX**: 累加器，运算时较多使用这个寄存器，有些指令规定必须使用它。
- ② **BX**: 基址寄存器，除了存放数据，它经常用来存放一段内存的起始偏移地址。
- ③ **CX**: 计数寄存器，除了存放数据，它经常用来存放重复操作的次数。
- ④ **DX**: 数据寄存器，除了存放数据，它有时存放32位数据的高16位。

寄存器介绍

❖ 通用地址寄存器

- 16位的8086处理器有4个16位的通用地址寄存器。
 - 它们的主要作用是存放数据的所在偏移地址，也可以存放数据。
 - **SP, BP, SI, DI**
 - 这4个寄存器不能被拆分使用。
- ① **SP**: 堆栈指针。这是一个专用的寄存器，存放堆栈栈顶的偏移地址。
 - ② **BP**: 基址指针。可以用来存放内存中数据的偏移地址。
 - ③ **SI**: 源变址寄存器。经常用来存放内存中源数据区的偏移地址
 - 所谓变址寄存器，是指在某些指令作用下它可以自动地递增或递减其中的值。
 - ④ **DI**: 目的变址寄存器。经常用来存放内存中目的数据区的偏移地址，并在某些指令作用下可以自动地递增或递减其中的值。

寄存器介绍

❖ 段寄存器

- 16位80x86处理器有4个16位的段寄存器
- 分别命名为 CS, SS, DS, ES
- 它们用来存放4个段的段基址

- ① CS: 代码段寄存器, 用来存放当前正在执行的程序段的段基址。
- ② SS: 堆栈段寄存器, 用来存放堆栈段的段基址。
- ③ DS: 数据段寄存器, 用来存放数据段段基址。
- ④ ES: 附加段寄存器, 用来存放另一个数据段的段基址。

寄存器介绍

❖ 指令指针寄存器

- **IP:** 指令指针寄存器，存放即将执行指令的偏移地址。

❖ 标志寄存器

- **FLAGS:** 存放CPU的两类标志。
- 状态标志：反映处理器当前的状态，如有无溢出，有无进位等。
 - 状态标志有6个：**CF、PF、AF、ZF、SF和OF**。见**表2-2**
- 控制标志：用来控制处理器的工作方式，如是否响应可屏蔽中断等
 - 控制标志有3个：**TF、IF和DF**

CS和IP

❖ 8086 CPU的工作过程的简要描述

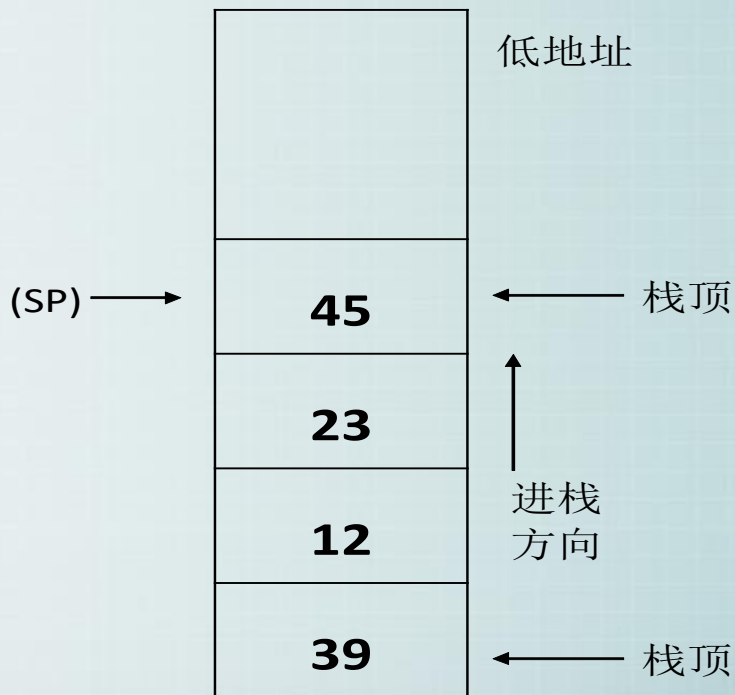
- ① 从CS:IP指向的内存单元读取指令，读取的指令进入指令缓冲器；
- ② $IP = IP + \text{所读取指令的长度}$ ，从而指向下一条指令；
- ③ 执行指令，转到步骤（1），重复这个过程。

❖ CS和IP的内容提供了CPU要执行指令的地址

- CS:IP
- 初始值为 FFFF:0000

堆栈

- ❖ 堆栈区是一个特殊的存储区
 - 它的末单元称为栈底
 - 数据先从栈底开始存放
 - 最后存入的数据所在单元称为栈顶
- ❖ 数据存取时采用后进先出的方式
- ❖ 当堆栈区为空时，栈顶和栈底是重合的。
- ❖ 数据在堆栈区存放时，必须以字存入，每次存入一个字，后存入的数据依次放入栈的低地址单元中。
- ❖ 栈指针SP每次减2(字节)，由栈指针SP指出当前栈顶的位置



← 2.3 存储器

- 一. 存储器
- 二. 存储器分段
- 三. 逻辑地址
- 四. CPU对内存的读写操作

存储器

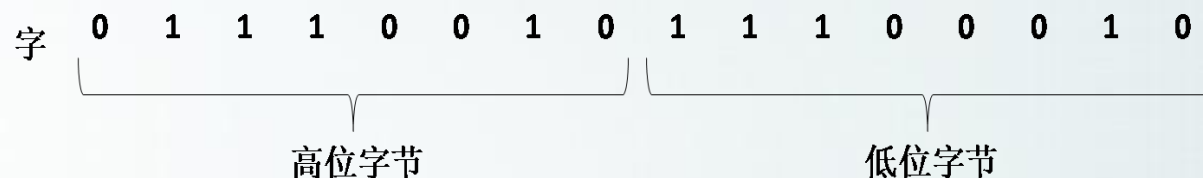
❖ 1. 基本存储单元

- 计算机存储信息的最小单位是一个二进制位 (bit)
- 8位二进制位组成一个字节(Byte)
- 80x86微机的内存储器以字节为基本存储单位，或叫基本存储单元
- 2个字节（16位）组成一个**字**(Word)
- 2个字（32位）称为**双字**。

存储器

❖ 2. 内存中字的存储

字与字节的对应关系



从31200H单元开始存放的**字数据**为A28FH，从31202H单元开始存放的字数据为1234H，分别记为：

(31200H) 字=A28FH

(31202H) 字=1234H

物理地址	存储单元
30000H	B8
30001H	23
30002H	72
30003H	E2
.....
31200H	8F
31201H	A2
31202H	34
31203H	12
.....
3FFFFH	C9

存储器分段

1. 分段

内存存在物理上并没有分段，分段只是CPU管理内存的方式。



分段示意图

存储器分段

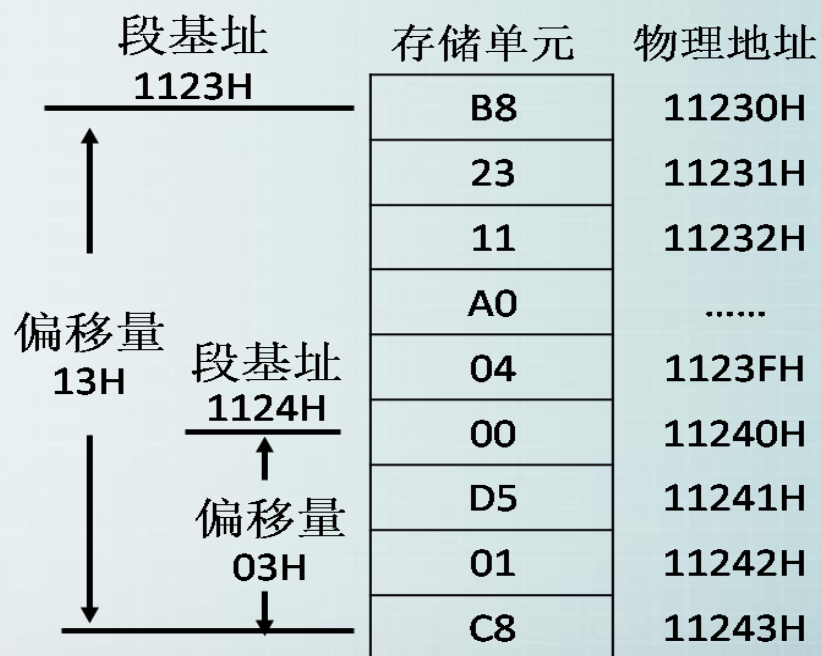
2.段的类型

- ❖ **代码段**—用于存放指令
 - 代码段段基址存放在段寄存器CS
- ❖ **数据段**—用于存放数据
 - 数据段段基址段地址存放在段寄存器DS
- ❖ **附加段**—用于辅助存放数据
 - 附加段段基址存放在段寄存器ES
- ❖ **堆栈段**—可用来保存数据、地址和系统参数
 - 堆栈段段基址存放在段寄存器SS

逻辑地址

❖ **逻辑地址**是用户编程时使用的地址，分为段地址和偏移地址两部分。

❖ **段地址**： **偏移地址**



PA与LA的对应关系

逻辑地址



如何描述从宿舍到教室的距离？

2	3	4	5
---	---	---	---

可以写下**4**位数据的纸条

2	0	0
---	---	---

3	4	5
---	---	---

可以写下**3**位数据的纸条

逻辑地址

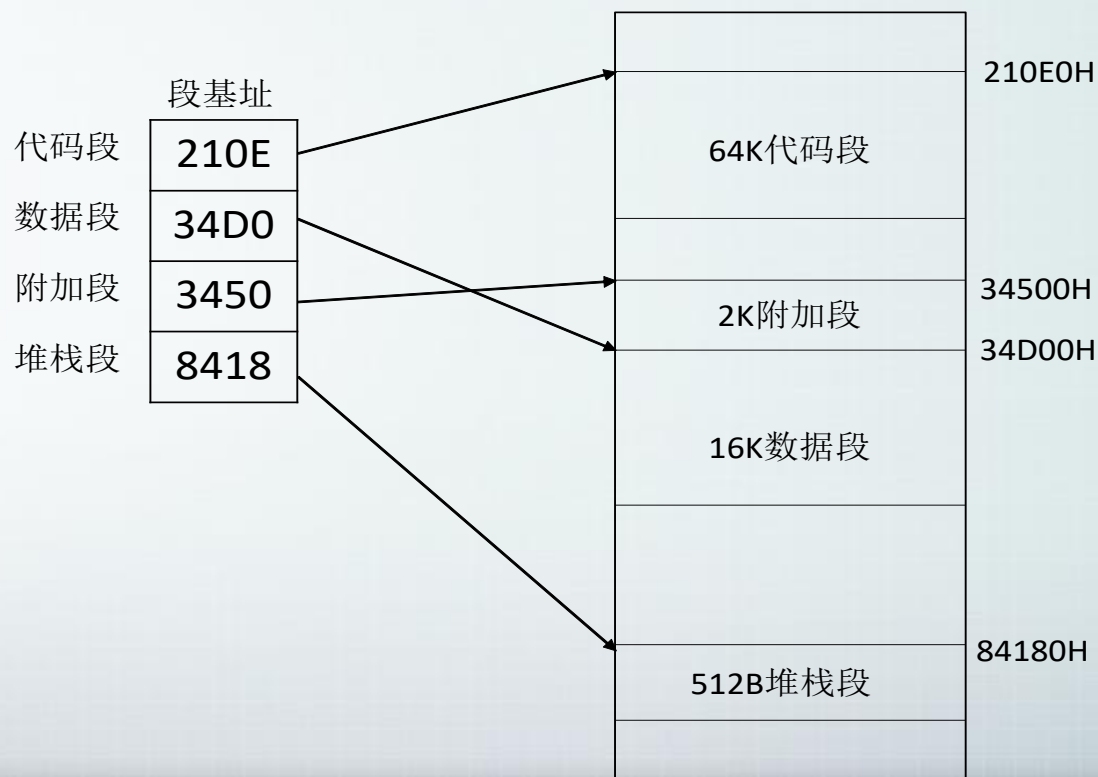
❖ 例题：

- 段基址为1896H，偏移地址为1655H。其物理地址为多少？

$$18960H + 1655H = 19FB5H$$

逻辑地址

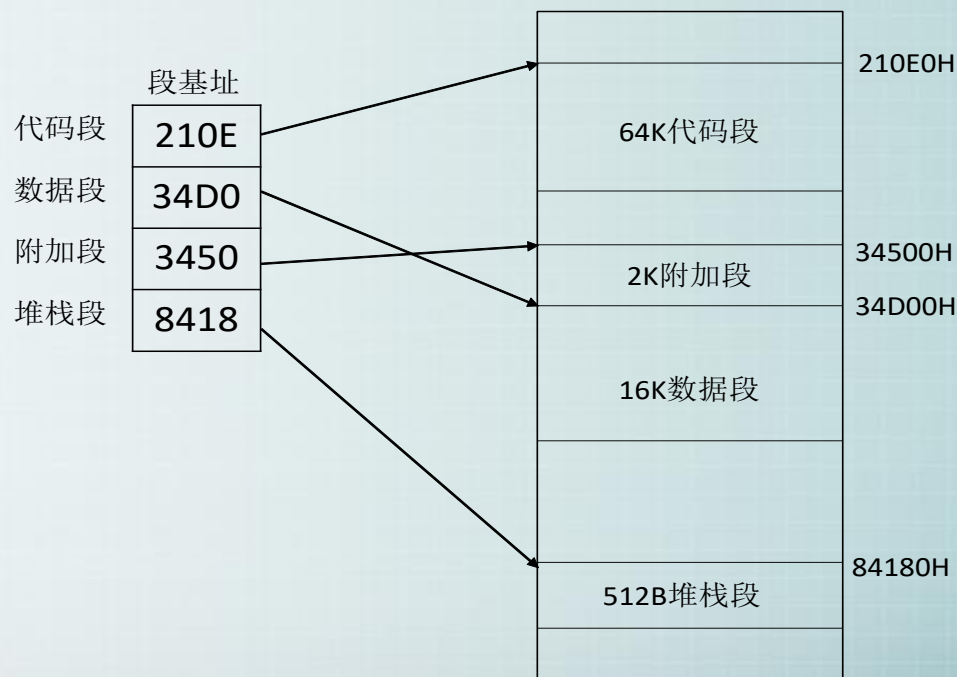
❖ 例：段基址与内存分段情况如图所示，观察各段的大小与分布，判断其地址范围标出每个段首地址和末地址。



逻辑地址

❖ 从图中看出

- 代码段有 **64KB**，它的地址范围在 **210E0H~310DFH**，已经达到段的最大范围。
- 附加段只有 **2KB**，地址范围在 **34500H~34CFFH**之间。
- 数据段为 **16KB**，其地址范围为 **34D00H~38CFFH**。可知数据段紧接着附加段的最后单元存放，而不必在附加段的**64KB**最大区域之外设置其他段。此方式也称为段重叠，可充分利用现有的存储空间。
- 堆栈段的空间最小，只有**512**个字节单元，它的地址范围是**84180H~8437FH**。



CPU对内存的读写操作

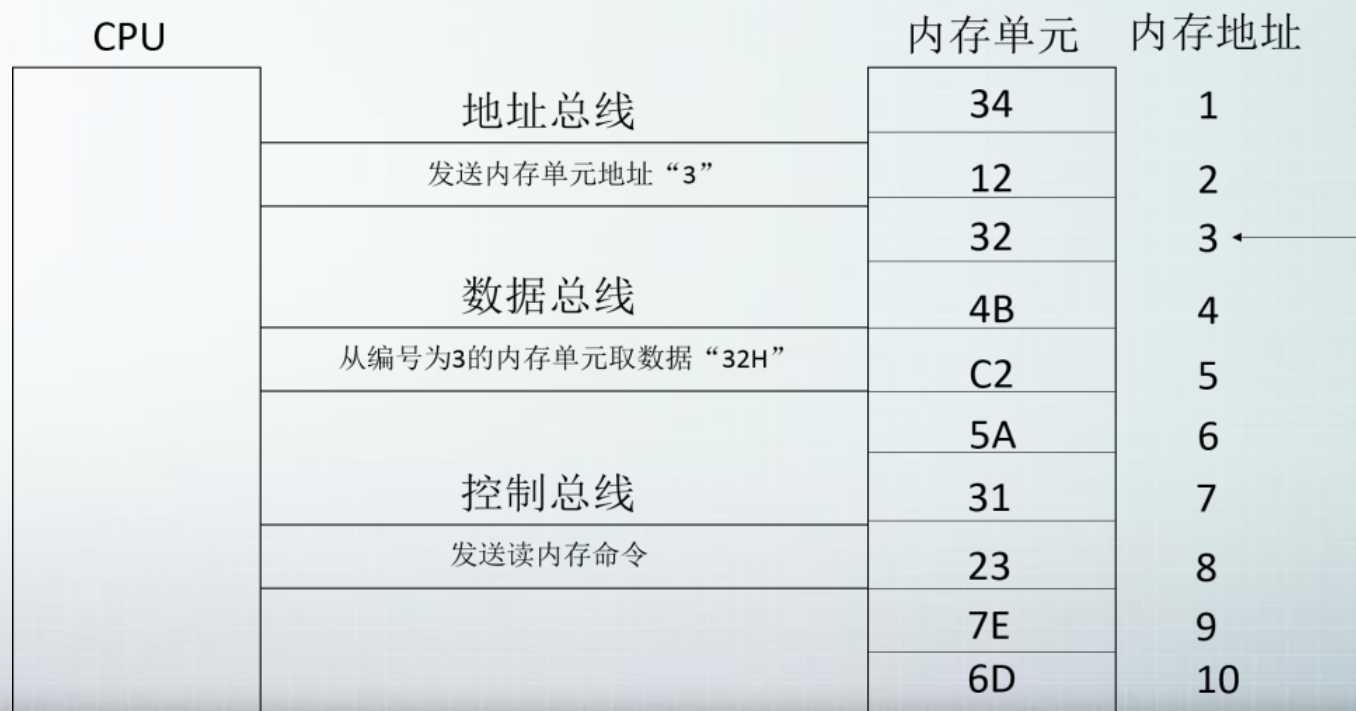
❖ CPU要想进行数据的读写，必须和外部器件（芯片）进行下面3类的信息交互。

- 存储单元的地址（地址信息）
- 器件的选择，读或写命令（控制信息）
- 读或写的数据（数据信息）

CPU对内存的读写操作

❖ CPU从地址为3的内存单元中读取数据的过程

- CPU通过地址线将要进行操作的内存单元地址“3”发出；
- CPU通过控制线发出内存读命令，选中存储器芯片，并通知它，将要从中读取数据
- 存储器将内存单元地址为“3”中的数据“32H”通过数据线送入CPU



❖ 写操作与读操作的步骤相似，如向地址为“3”的单元写入数据“FFH”。

- CPU通过地址线将要进行操作的内存单元地址“3”发出；
- CPU通过控制线发出内存写命令，选中存储器芯片，并通知它，要向其中写入数据。
- CPU通过数据线将数据“FFH”送入内存的地址为“3”的单元中。

CPU对内存的读写操作

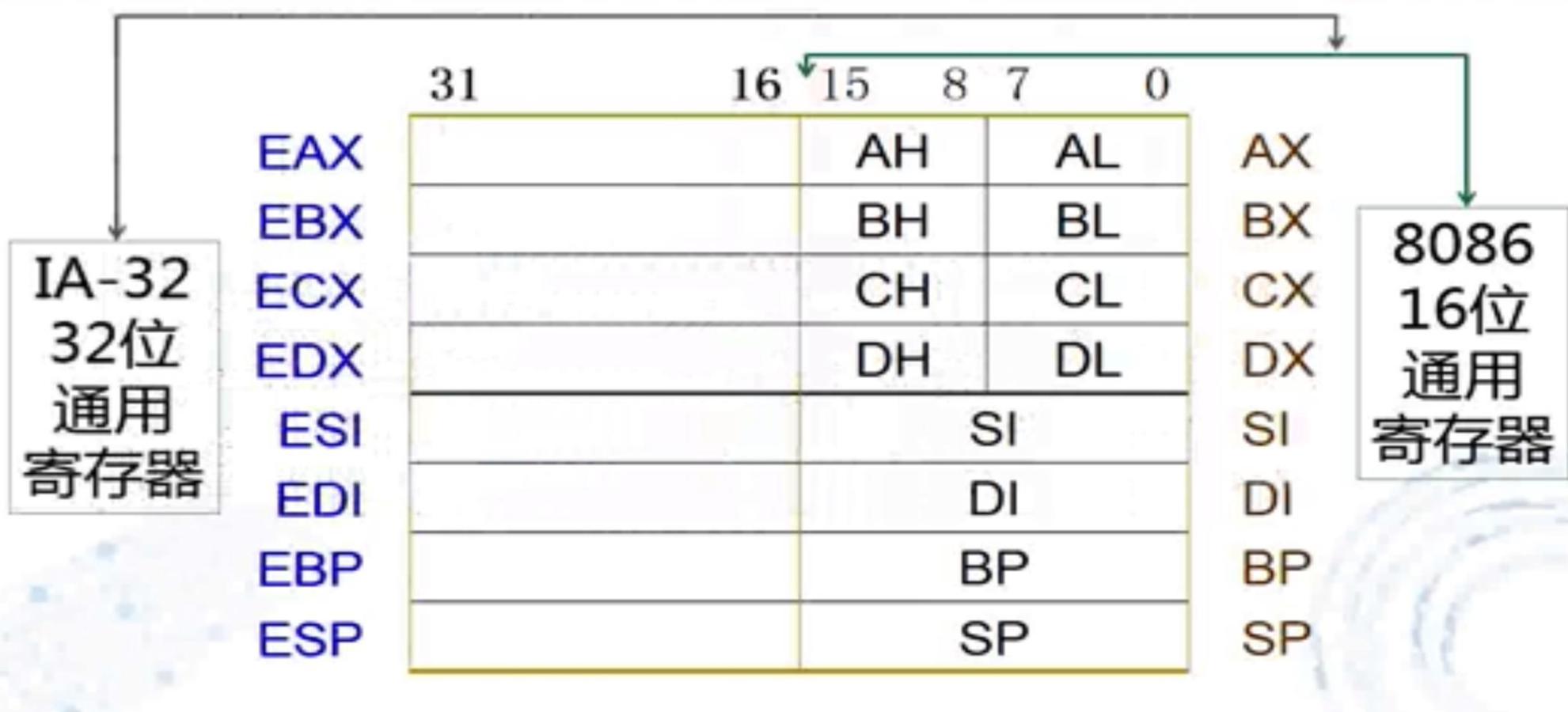
- ❖ 要让一个计算机或微处理器工作，应向它输入能够驱动它进行工作的电平信息（即机器码）。
- ❖ 对于8086 CPU，下面的机器码，能够完成从内存地址为“3”的单元读数据。
 - 机器码：101000010000001100000000
 - 含义：从内存地址为“3”的单元读取数据送入寄存器AX
- ❖ 机器码是01串，难以记忆和书写，用汇编指令来表示，情况如下。
 - 机器码：10100001 00000011 00000000
 - 对应的汇编指令：mov AX, [3]
 - 含义：从内存地址为“3”的单元读取数据送入寄存器AX

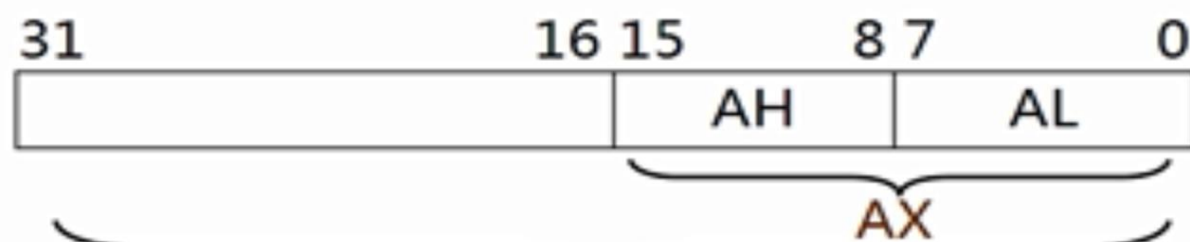
← 2.4 外部设备和接口

- ❖ 外部设备是计算机系统的不可缺少的重要组成部分。通过输入设备把程序和数据输入计算机主机（CPU和内存），通过输出设备把结果输出给用户或保存起来。
- ❖ 根据不同用途，接口中的寄存器（端口）分为以下3类。
 - 数据端口
 - 控制端口
 - 状态端口
- ❖ CPU与I / O接口中端口的信息传输也都是通过数据总线进行的。

IA-32 32位通用寄存器

- ❖ IA-32为Intel Architecture 32-bit简称，即英特尔32位体系架构
- ❖ 从1985年面世的80386直到Pentium 4，都是使用IA-32体系结构的处理器。





既是一个整体
又可独立使用

EAX
EBX

ECX

