

第二章——微处理器结构

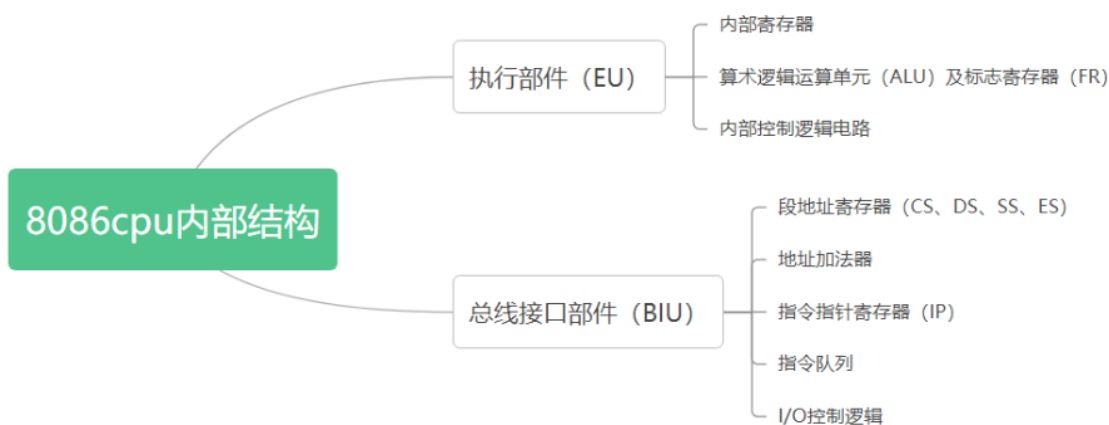
1. 8086和8088 的地址线和数据线情况

8086和8088的数据线和地址线都是复用的

- 8086有16位数据总线，有20根地址总线，有16个地址/数据复用引脚（最大寻址空间是1MB）
- 8088有8位数据总线，有8根地址/数据复用引脚

2. 什么是编程结构，分哪两部分

- 从功能上看，8086分为，总线接口部件BIU，执行部件EU



CSDN @shanzil

3. 段地址不会特别考，但是要会从逻辑地址到物理地址的转换

内存的分段是逻辑分段，不是物理段，每个内存单元具有唯一的物理地址，但可能有多个逻辑地址，逻辑段的大小、位置都可以改变，一个16位的计算机逻辑段的默认容量是64kb

- **段地址说明逻辑段在主存中的起始位置**
- 8086规定段地址必须是模16地址: xxxx0H
- 省略低4位0000B，段地址就可以用16位数据表示，就能用16位段寄存器表达段地址

➤ 将逻辑地址中的段地址左移4位，加上偏移地址就得到20位物理地址

➤ 一个物理地址可以有多个逻辑地址

	逻辑地址	1460:100、1380:F00
	物理地址	14700H 14700H
段地址左移4位	→	14600H 13800H
加上偏移地址	→	+ 100H + F00H
得到物理地址	→	14700H 14700H

由于这个省略低4位原因，所以要得到真正的段地址时应该先*16

- 设某操作数存放在数据段，DS=250AH，数据所在单元的偏移地址=0204H。则该操作数所在单元的物理地址为：

$$250AH \times 16 + 0204H = 252A4H$$

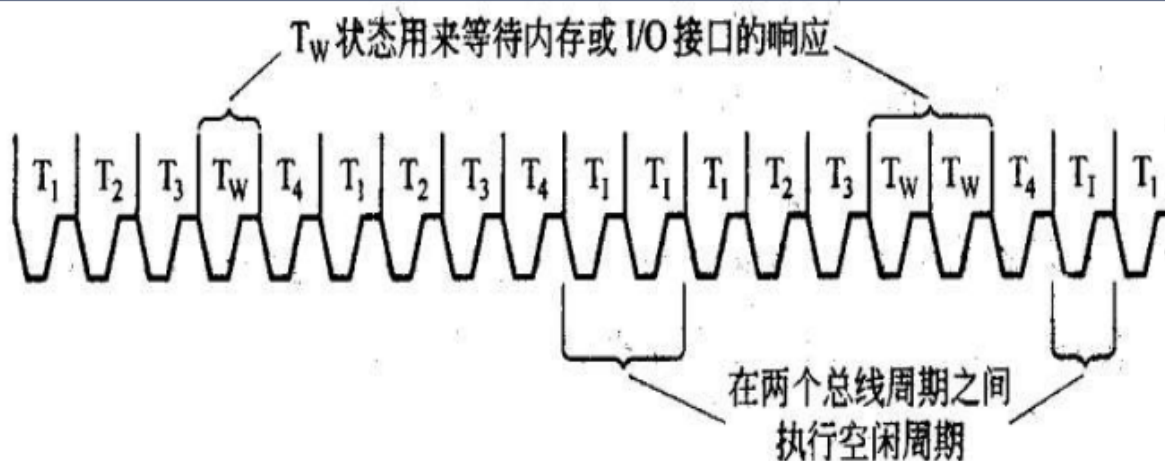
段地址：操作系统给出

偏移地址：指令给出

合成后的物理地址

4. 什么是总线周期？一个总线周期有几个状态？如果要是处于等待状态，应该是哪个时刻？

- 总线周期：CPU完成一次访问内存(或接口)操作所需要的时间
 - 指令周期：CPU从内存取出一条指令并执行完这条指令的时间总和
 - CPU周期：又称机器周期(总线周期)，CPU访问内存所花的时间较长，因此用CPU从内存读取一条指令字所需的的最短时间来定义
 - 时钟周期：通常称为节拍脉冲或T周期时钟周期个CPU周期包含若干个时钟周期
-
- 取指令或传送数据，就需要CPU的总线接口部件执行一个总线周期
 - 在8086/8088中，一个基本的总线周期由4个时钟周期组成，时钟周期是CPU的基本时间计算单位，它由计算机主频决定。4个时钟周期分别称为4个状态，即T1状态，T2状态，T3状态，T4状态



- 所以等待的时候是在T3和T4之间吗（如果有的话）

5. 最小最大工作模式是什么

- 就是在系统中只有8086或者8088一个微处理器。在这种系统中，所有的总线控制信号都直接由8086或8088产生，因此，系统中的总线控制电路可减到最少
- 在最大模式系统中，总是包括有两个或多个微处理器，其中一个主处理器就是8086或者8088，其他的处理器称为协处理器，协助主处理器工作。



6. 五种基本操作

- 复位操作

复位完了问初始状态是什么（各寄存器的状态和物理地址等）

86、88CPU的reset引脚用来使CPU复位和启动，被复位后，PSW、DS、ES、SS和其他寄存器被清零，指令队列也被清零，段寄存器CS和指令指针IP分别被初始化为0FFFFH和0000H

表 2.6 复位时各内部寄存器的值

标志寄存器	清 零
指令指针(IP)	0 0 0 0 H
CS 寄存器	F F F F H
DS 寄存器	0 0 0 0 H
SS 寄存器	0 0 0 0 H
ES 寄存器	0 0 0 0 H
指令队列	空
其他寄存器	0 0 0 0 H

- 总线操作

总线读和总线写，最典型的区别是什么？

二者状态区别在T2

- 中断

中断的分类？什么是中断向量？给出已知条件求中断向量？

1. 中断分为硬件中断（非屏蔽中断和可屏蔽中断）和软件中断

第三章——汇编语言

1. 会取址方式，寻址方式，寻址方式会专门考，指令不会专门考，会常用指令就行，会写程序就好

具体知识点见速成笔记寻址方式吧，下面是一些题目

这一块题需要记忆基本的寄存器，8个通用寄存器，4个段寄存器

8个通用：

- 数据寄存器 (AX, BX, CX, DX)
- 地址指针寄存器 (SP, BP)
- 变址寄存器 (SI, DI)

4个段：

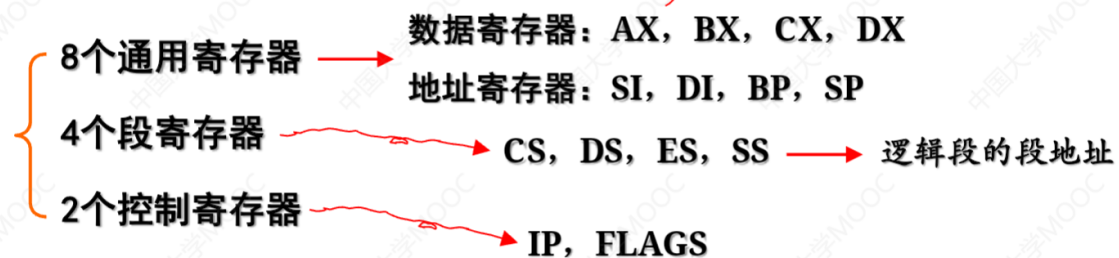
- CS
 - 代码段寄存器，存放代码段的段基地址。
- DS
 - 数据段寄存器，存放数据段的段基地址。
- ES
 - 附加段寄存器，存放数据段的段基地址。
- SS
 - 堆栈段寄存器，存放堆栈段的段基地址

段寄存器的值表明相应逻辑段在内存中的位置

含14个16位寄存器，按功能可分为三类

可以分别拆分为两个8位寄存器：

AL, AH, BL, BH,
CL, CH, DL, DH



3.1 寻址方式的题

1. 第一题

- 偏移地址=1400H 段基地址=SS=1400H
- ① 设DS=1300H, SS=1400H, BP=1500H, SI=0100H。则指令：
MOV AX, [BP+SI-200H] 的源操作数的物理地址为 (15400) H。

是基址变址寻址，首先确定逻辑段，由基址寄存器决定，BP，所以是堆栈段，那么堆栈段是SS，故逻辑地址是1400H，计算偏移地址是1400H，所以物理地址=1400H+1500H=15400H

2. 第二题 寻址方式判断

- ① 对源操作数而言，下列常用寻址方式中，（ C ）属于立即寻址，（ B, E ）属于寄存器间接寻址，（ F ）属于寄存器相对寻址，（ A ）属于直接寻址。

A. MOV AX, [2000H]

B. MOV AX, [BX]

C. MOV AX, 2000H

D. MOV AX, BX

E. MOV AX, [DI]

F. MOV AX, [BX+6]

3.2 常见的一些指令

MOV、IN/OUT、jmp指令 以及一些移位指令

主要是芯片程序会写，汇编语言程序不用可以去学

第四章——存储器

4.1 存储器的分类

- 按制造工艺

◆ 双极型：速度快、集成度低、功耗大

◆ MOS型：速度慢、集成度高、功耗低

- 按使用属性

◆ 随机存取存储器RAM：可读可写、断电丢失

◆ 只读存储器ROM：正常只读、断电不丢失

- RAM和ROM的具体细分



- RAM具体划分

	组成单元	速度	集成度	应用
SRAM	触发器	快	低	小容量系统
DRAM	极间电容	慢	高	大容量系统
NVRAM	带微型电池	慢	低	小容量非易失

- ROM具体划分

- ◇掩膜ROM：信息制作在芯片中，不可更改
- ◇PROM：允许一次编程，此后不可更改
- ◇EPROM：用紫外光擦除，擦除后可编程；并允许用户多次擦除和编程
- ◇EEPROM（E²PROM）：采用加电方法在线进行擦除和编程，也可多次擦写
- ◇Flash Memory（闪存）：能够快速擦写的EEPROM，但只能按块（Block）擦除

4.2 芯片图

给出已知芯片，给出目标容量，会画图，会写每个芯片的地址修改范围，例题如下

用2732（4K*8）扩展16K*8的ROM

（1）共需多少片2732？

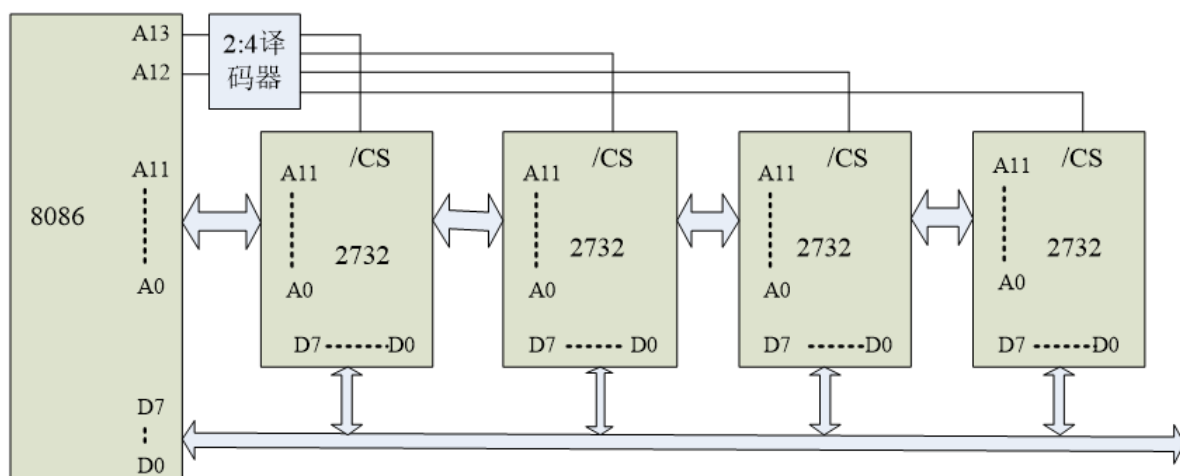
（2）每个芯片需要多少根地址线进行片内译码？

（3）画出硬件扩展电路并写出每个芯片的地址覆盖范围。地址从0000H开始

答案如下

1. 4片
2. 12
3. 地址覆盖范围为:

0000H-0FFFH, 1000H-1FFFH, 2000H-2FFFH, 3000H-3FFFH



第六章

6.1 CPU与外设

CPU通过I/O接口与外设进行数据交换，有哪几种方式？每种方式的优缺点（5种）和各自的适用范围是什么？

- 无条件传输，查询传输，中断传输，DMA方式和I/O处理机方式
- 无条件传输: 适用于简单设备，传输前外设必须就绪。慢速外设需与CPU保持同步
- 查询传输: 工作可靠，适用面宽，但传送效率低
- 中断传输: 效率更高，可以处理随机请求，可以处理复杂事务。可与CPU并行工作，但每次传送需要大量额外时间开销。
- DMA方式: DMAC控制，外设直接和存储器进行数据传送，适合简单、大量、快速数据传送I/O处理机方式: 功能强大，但成本较高

6.2 I/O端口的地址编制方式

一共有2种，一种是单独编制，一种是统一编制。那么8086用的是哪一种编制呢

接口电路占用的I/O端口有两类编排形式

◇ I/O端口单独编址

◆ I/O地址空间独立于存储地址空间

◆ 如8086/8088

◇ I/O端口与存储器统一编址

◆ 它们共享一个地址空间

◆ 如M6800

6.2.1 8086 关于I/O编址

- 8086用的是单独编址
- 8086用于寻址外设端口的地址线为16条端口最多为 $2^{16}=65536$ (64K), 端口号为0000H~ FFFFH
- 每个端口用于传送一个字节的外设数据

6.2.2 I/O寻址方式

- 8088/8086的端口有64K(2^{16} 次方)个, 无需分段, 设计有两种寻址方式
- 直接寻址: 只用于寻址00H~ FFH前256个端口, 操作数i8表示端口号
- 间接寻址: 可用于寻址全部64K个端口, DX寄存器的值就是端口号
- 对大于FFH的端口只能采用间接寻址方式

6.3 芯片编程

所有芯片的初始化程序都要求会写, 8255还要求会写应用程序, 中断也是一样的, 要求会写初始化程序, 包括定时计数器, 串行接口多一个, 异步和同步的优缺点(涉及到一些概念)

6.3.1 IN和OUT指令

对于输入输出, 只要是内部的基本记住是AL和AX

对于PORT, 如果端口地址是8位, 可以直接给出, 如果是16位, 必须由DX给出,
IN acc, PORT
对于acc, 可以是AL或AX
OUT PORT, acc

◇ 输入指令 (IN: 将外设数据传送给CPU内的AL/AX)

IN AL,i8 ;字节输入

IN AL,DX ;字节输入

IN AX,i16 ;字输入

IN AX,DX ;字输入

演示

◇ 输出指令 (OUT: 将CPU内的AL/AX数据传送给外设)

OUT i8,AL ;字节输出

OUT DX,AL ;字节输出

OUT i16,AX ;字输出

OUT DX,AX ;字输出

演示

一些关于AX, AL的解释

◇ 如果输入输出一个字节, 利用AL寄存器

◇ 如果输入输出一个字, 利用AX寄存器

◇ 输入一个字, 实际上是从连续两个端口输入两个字节, 分别送AL (对应低地址端口) 和AH (对应高地址端口)

◇ 输出一个字, 实际上是将AL (对应低地址端口) 和AH (对应高地址端口) 两个字节的内容输出给连续两个端口