

《微机原理与接口》


第6章 中断和8259A

教师：苏曙光
华中科技大学软件学院

- 教学内容
 - 第1节 中断概念
 - 第2节 中断源识别和优先权
 - 第3节 8259A性能和结构
 - 第4节 8259A的工作方式
 - 第5节 8259的中断操作和命令
 - 第6节 8086的中断系统



第1节 中断的概念

- 中断的概念
 - CPU的机制或过程
 - CPU在正常运行程序时，由于内部/外部事件引起CPU中断正在运行的程序，转去执行为该事件预先安排的服务程序，服务完毕后，再返回原来的程序继续执行。
 - 中断的作用和应用
 - ①同步操作/并行操作
 - ②实时处理
 - ③故障或异常处理
- 

- 中断源和其分类

- 中断源：引起中断的事件或发出中断的外设。

- 分类

- ◆ 外部中断

- 可屏蔽中断（INTR引脚）

- ▲ 外设

- 不可屏蔽中断（NMI引脚）

- ▲ 电源，内存错误等

- ◆ 内部中断

- CPU内部

- 软件中断（指令中断）

- 异常（指令执行时发生错误）

- 中断系统的功能

- 能发现和识别中断源，实现中断服务。

- ①实现中断和从中断返回

- ◆ 外设能发出中断请求；

- ◆ CPU能决定是否响应中断(响应中断的条件)

CPU响应中断的条件

- ①必须执行完现行指令
 - 运行到最后一个总线周期的最后一个时钟状态（T4），CPU检测INTR引脚是否有中断请求，有则响应中断，进入中断总线周期。
- ②开中断状态
 - STI指令
- ③无DMA操作

- 中断系统的功能
 - 能发现和识别中断源，实现中断服务。
 - ①实现中断和从中断返回
 - ◆ 外设能发出中断请求；
 - ◆ CPU能决定是否响应中断
 - ◆ 中断响应过程
 - 保护现场-清除中断请求-中断服务-恢复现场-返回主程序

中断响应过程

- ①关中断——发出响应信号INTA，并自动关中断。
- ②保留断点——主程序下一条指令地址（CS：IP）入栈。
- ③保护现场——断点处各寄存器的内容入栈。
- ④根据中断入口地址，转入相应中断服务程序。
- ⑤恢复现场——把现场和断点的内容出栈。
- ⑥中断返回——在中断服务程序的最后执行IRET，并开中断。
- 注：①②④由CPU自动完成；
③⑤⑥由程序完成。

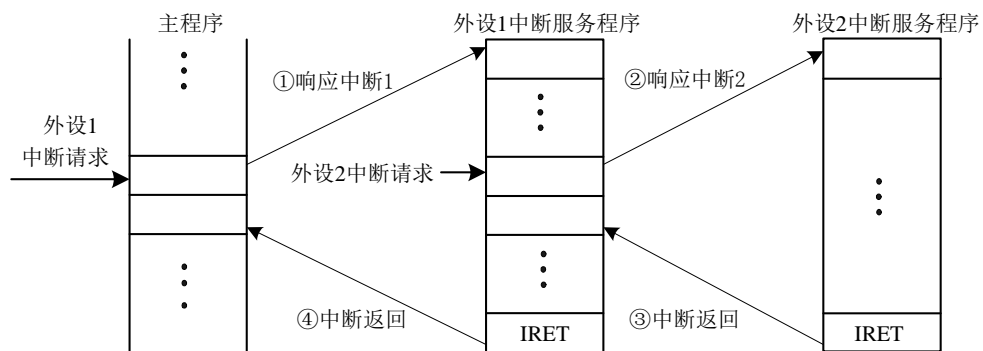
● 中断系统的功能

- 能发现和识别中断源，实现中断服务。
- ①实现中断和从中断返回
 - ◆ 外设能发出中断请求；
 - ◆ CPU能决定是否响应中断
 - ◆ 中断响应过程
 - 保护现场-清除中断请求-中断服务-恢复现场-返回主程序
- ②实现优先权排队
 - ◆ 当有多个中断申请时，先响应优先级高的中断
- ③能实现中断嵌套。
 - ◆ 即高优先级中断能中断低优先级的中断。

中断嵌套

● 例子

■ 外设2的优先级 > 外设1的优先级

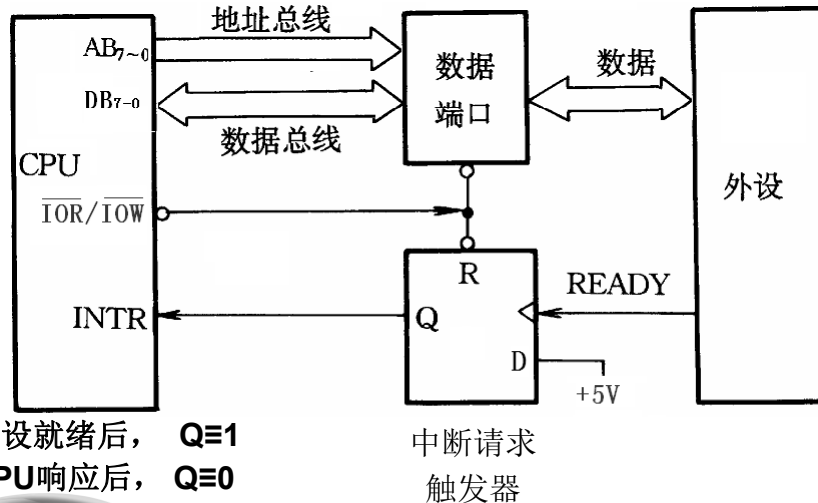


微机中断的基本硬件机制

- 1、外设：能发出且保持中断申请直到其响应为止。
- 2、中断接口：能屏蔽或开放某个外设的中断请求。
- 3、CPU：能开放（使能）或关闭（失能）系统的中断机制。
- 4、能识别中断源和管理中断优先级
- 5、确定发现中断的时机（T4态）

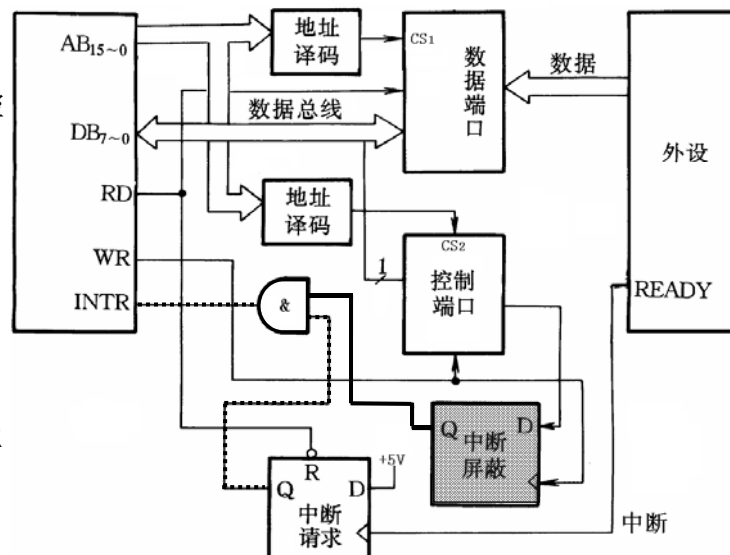
1、外设：能发出且保持中断申请直到其响应为止。

- 措施：每个中断源配置一个中断请求触发器（1bit）。



2. 中断接口：能屏蔽或开放某个外设的中断请求

- 中断请求触发器的 Q 端不再直接连接 $INTR$ ，而是受到控制。
- 每个中断源配置一个中断屏蔽触发器（1bit）。
- 增设一个控制端口
- 观察与思考: (1)两个端口; (2)两个D触发器; (3) RD/WR 信号连线和作用; (4)如何屏蔽外设的中断请求?

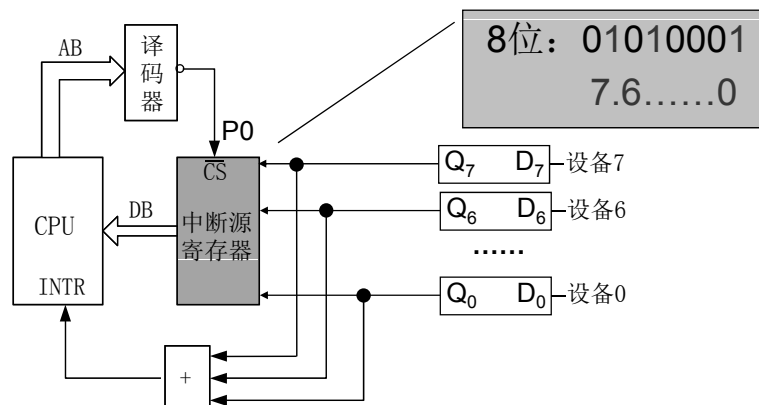


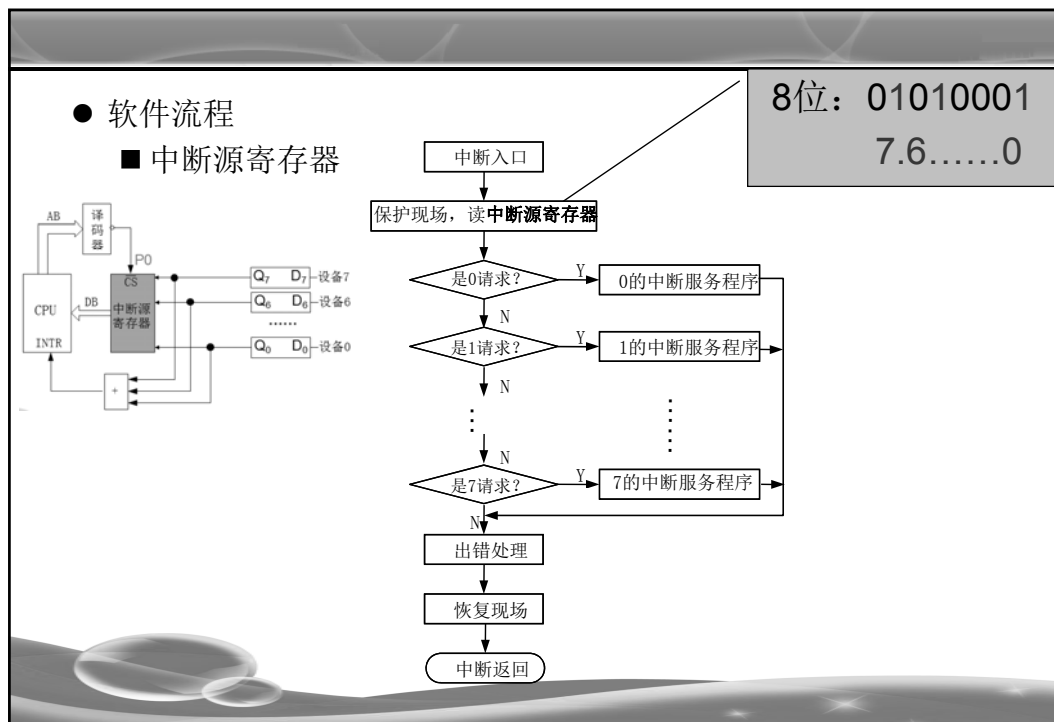
4、能识别中断源和管理中断优先级

- 识别中断源和中断优先级
 - 所谓优先级，是指有多个中断源同时提出中断请求时，微处理器响应中断的优先次序。
- 实现方法
 - 软件方法（查询中断）
 - 硬件方法（向量中断）

软件方法（查询中断）

- 硬件原理
 - 将8个外设中断请求触发器组合为中断源寄存器(端口地址P0)
 - 将8个中断请求信号输入或门，其输出连接INTR。





● 程序实现（屏蔽法）

```

IN      AL, [20H] ; 输入中断源寄存器的数据
TEST    AL, 01H   ; 检查位0（电源故障）是否有中断请求
JNE     PWF       ; 有，则转至电源故障处理程序
TEST    AL, 02H   ; 检查位1(磁盘)是否有中断请求
JNE     DISS      ; 有，转至磁盘服务程序
TEST    AL, 04H   ; 检查位2(键盘)是否有中断请求
JNE     KEYBRD    ; 有，转至键盘服务程序
.....
  
```

注意：JNE指令：当ZF = 0跳转。

(位与的结果不为0，则ZF = 0，否则ZF=1)

● 程序实现（移位法）

```
IN      AL, [20H]    ; 输入中断源寄存器的数据
RCL     AL, 1        ; 左移1位，带进位
JC      BIT7         ; 有进位跳转：位7对应中断服务程序
RCL     AL, 1        ; 左移1位，带进位
JC      BIT6         ; 有进位跳转：位6对应中断服务程序
.....
RCL     AL, 1        ; 左移1位，带进位
JC      BIT0         ; 有进位跳转：位0对应中断服务程序
```

● 软件查询方法特点

- ①外设的优先权为查询次序
 - ◆ 优先级最高的最先被查询。
- ②硬件简单
- ③缺点
 - ◆ 进入中断服务时间较长。

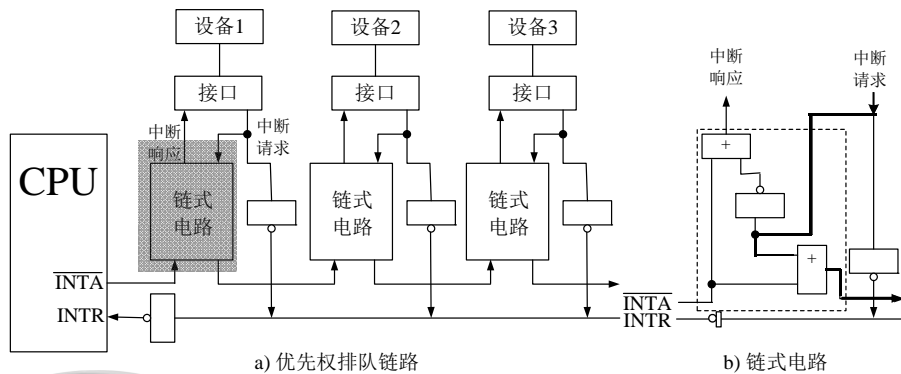
● 软件查询方法改进

- 用硬件实现查询和优先权管理
- 链式电路

软件查询方法改进——用硬件实现查询和优先权管理

- 链式优先权排队

- 排在左端的中断源优先级更高。
- 多个中断发生时，优先响应最左端的中断申请。



- 通过硬件实现查询和优先级。本质还是查询方法，中断响应和进入较慢。

- 向量中断——用硬件方法确定中断源及优先权

- 给每一个外设都预先指定一个中断类型码（N）。当CPU识别出外设有请求中断并予以响应时，中断控制逻辑把N送入CPU，CPU据此计算中断服务程序的入口地址，并转入中断服务程序。

- 特点：

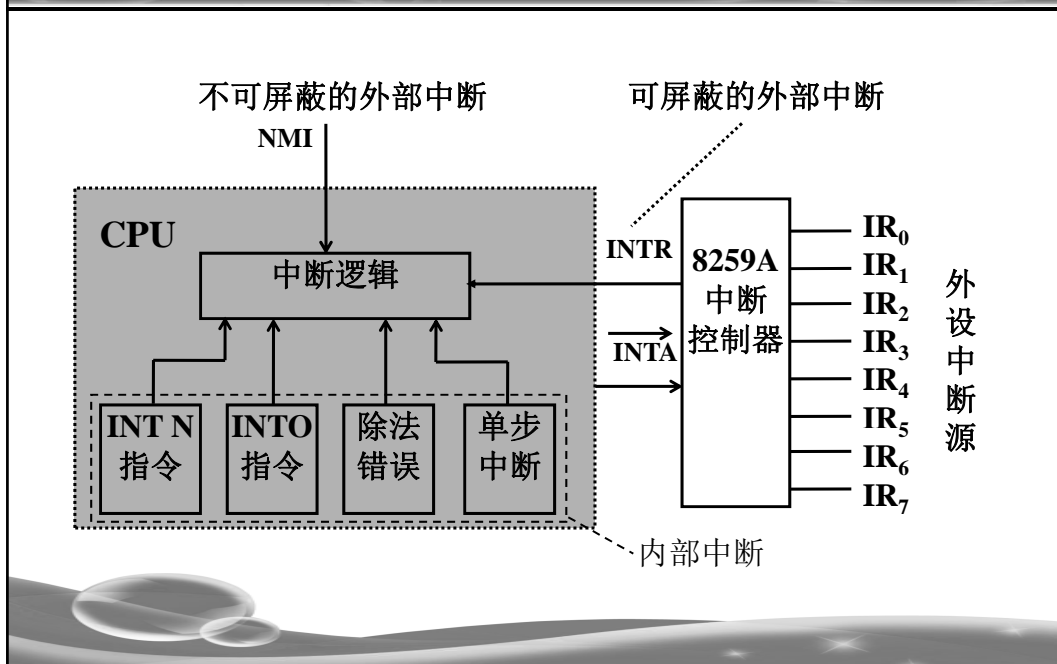
- ◆ 响应速度快

- ◆ 硬件复杂

- ◆ 典型芯片：8259

第2节 8086的中断机制

8086中断类型



8086的中断类型

- 外部中断 (Interrupt)
 - (1) 可屏蔽中断：来自外设，受IF控制， $N = 8 \sim 255$
 - (2) 非屏蔽中断：处理系统意外或故障，不受IF控制， $N=2$
 - ◆ **NMI = Non Maskable Interrupt**
- 内部中断 (异常, Exception)：8088内部执行程序错误
 - (1) DIV或IDIV指令
 - (2) INT指令
 - (3) INTO指令
 - (4) 单步中断
 - 为用户提供发现、调试并解决程序执行异常的途径
 - ◆ 例如BIOS和DOS功能调用，DEBUG

陷阱标志位TF=1时，每条指令执行完引起中断。 $N=1$

- 中断源的优先权

- 软件中断

- ◆ 除法错中断

- ◆ 指令中断

- ◆ 溢出中断

- 非屏蔽中断

- 可屏蔽中断

- 单步中断



8086中断向量表

- 中断向量

- 中断服务程序的入口地址 CS:IP

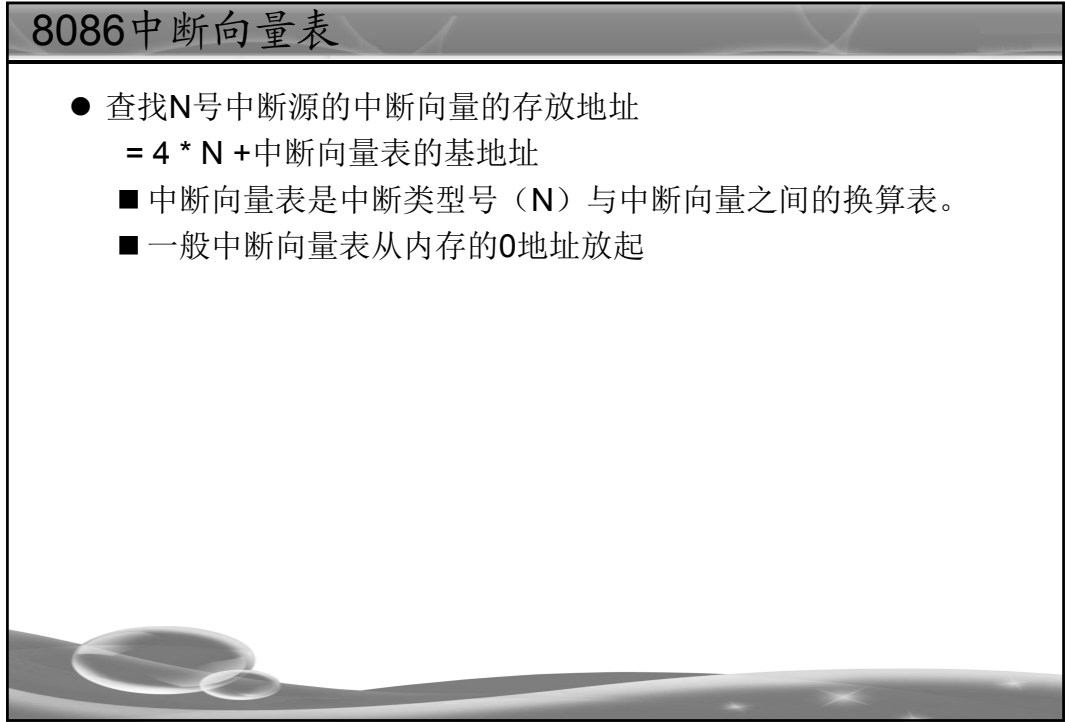
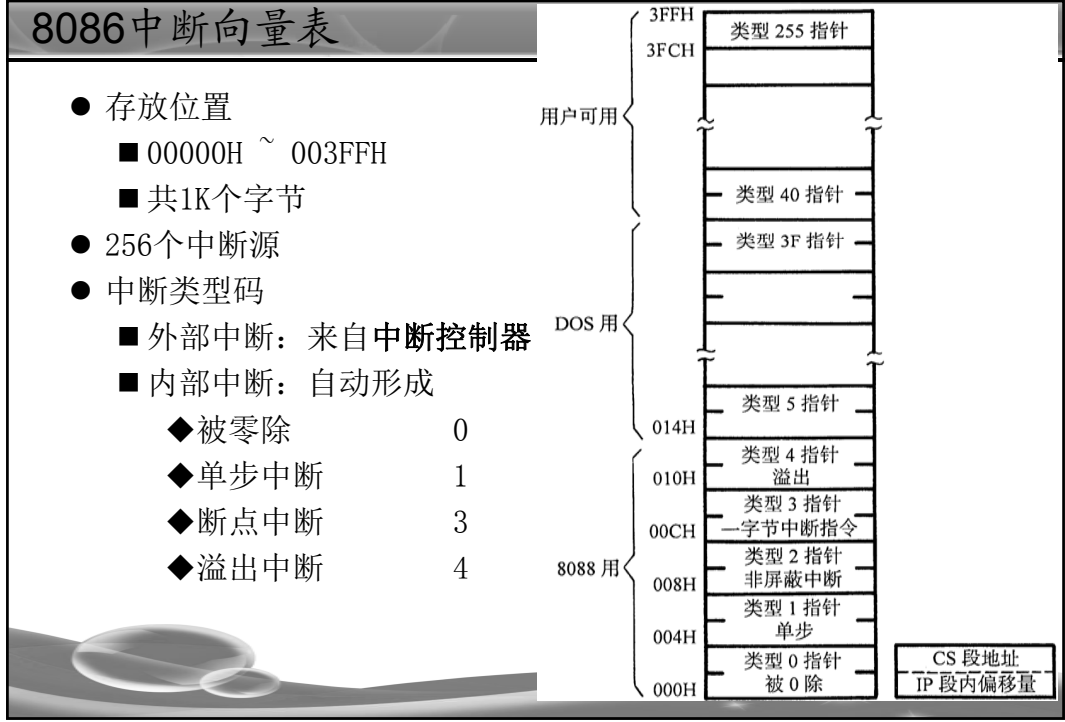
- ◆ 段基址CS: 偏移地址IP

- ◆ 每个向量占4字节。

- 中断向量表

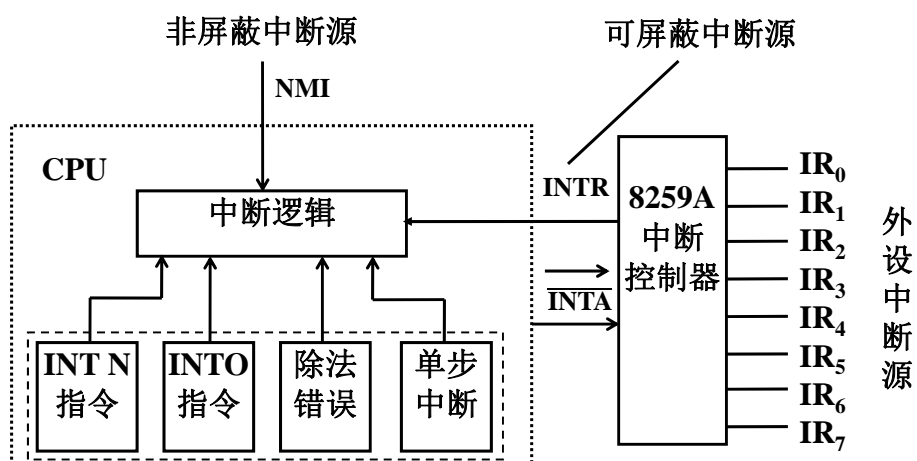
- 中断向量表是按序（中断源编号N）存放中断向量的表。

- 中断向量表存放在内存中（基址0）。



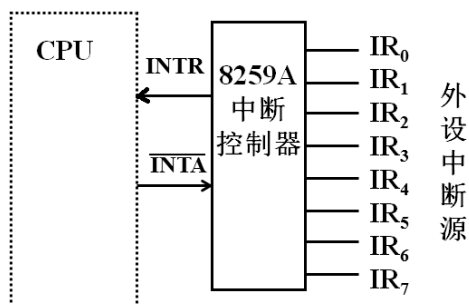
第3节 8259A性能和结构

● 8088/8086中断源



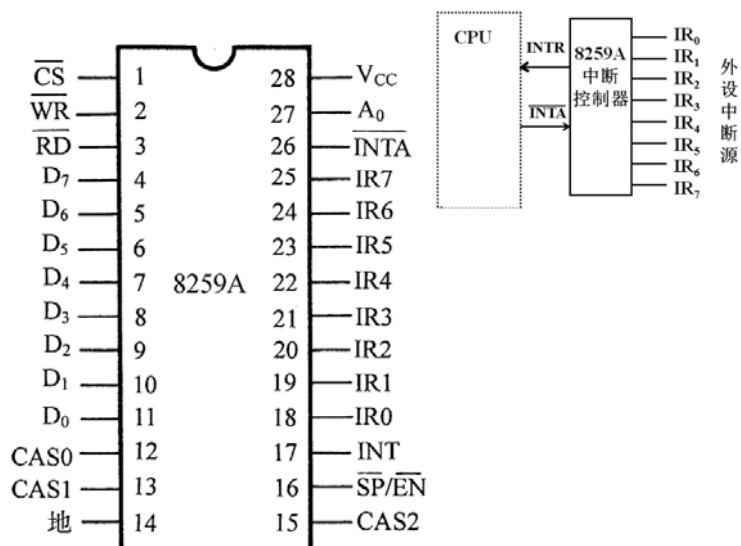
● 8259A的基本功能

- 1.能接收8个中断请求（可扩展到64级）
- 2.能提供中断类型号N（中断矢量）
- 3.能屏蔽和开放外设的中断请求
- 4.支持多种方式的优先级管理

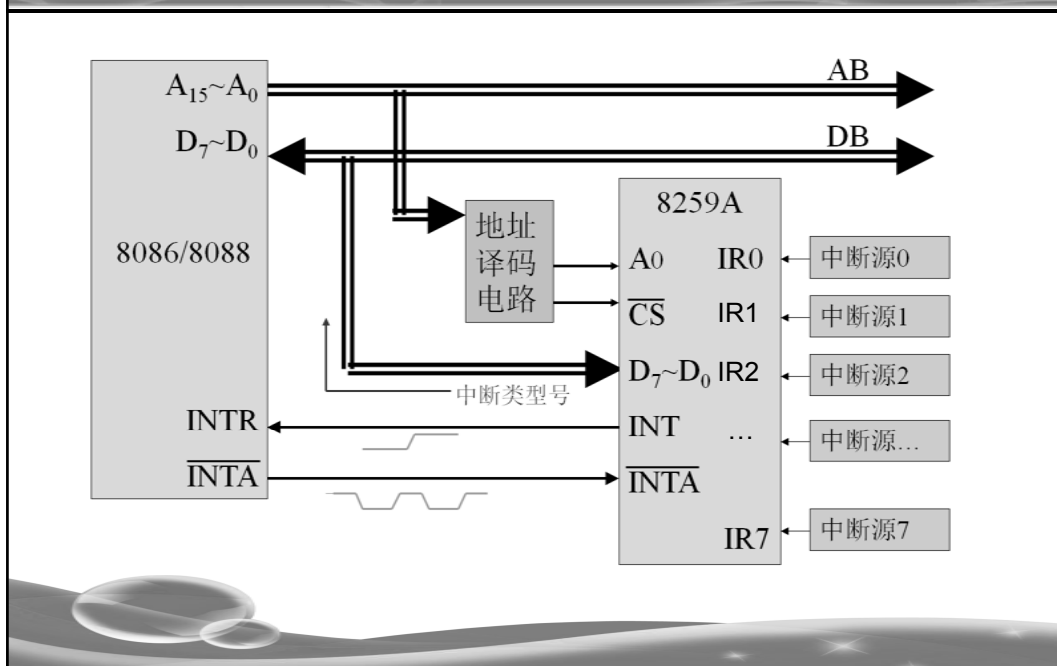


8259A外部引脚

- 28脚
- 介于CPU与外设之间



8259A在8088中断系统的连接



● 外部引脚

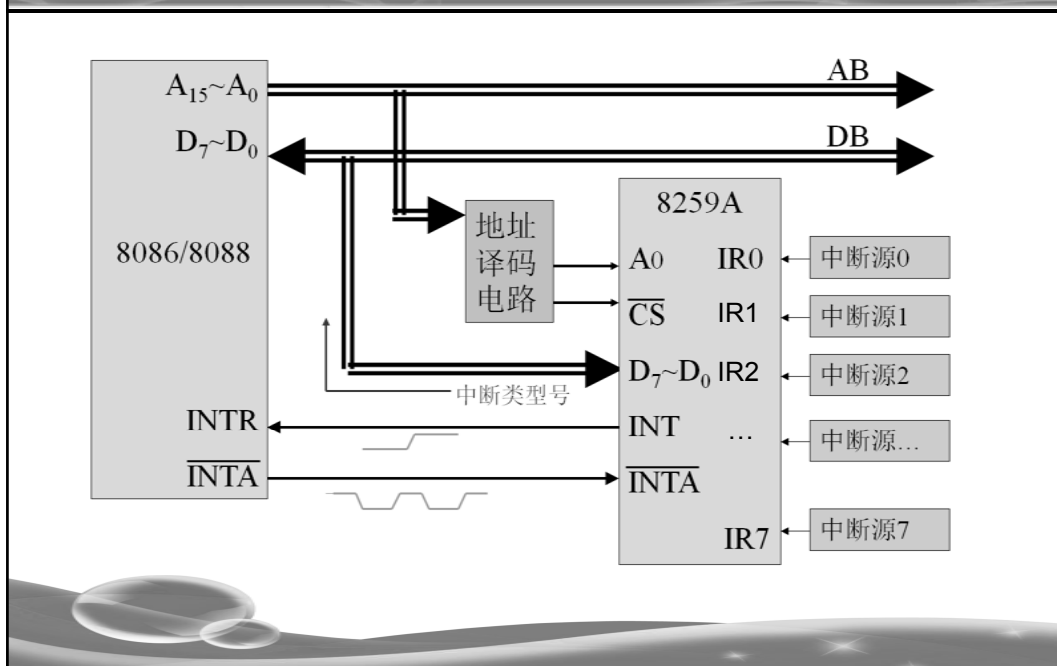
■ 面向CPU的引脚

◆D7~D0, $\overline{\text{CS}}$, A0, $\overline{\text{RD}}$,
 $\overline{\text{WR}}$, INT, $\overline{\text{INTA}}$

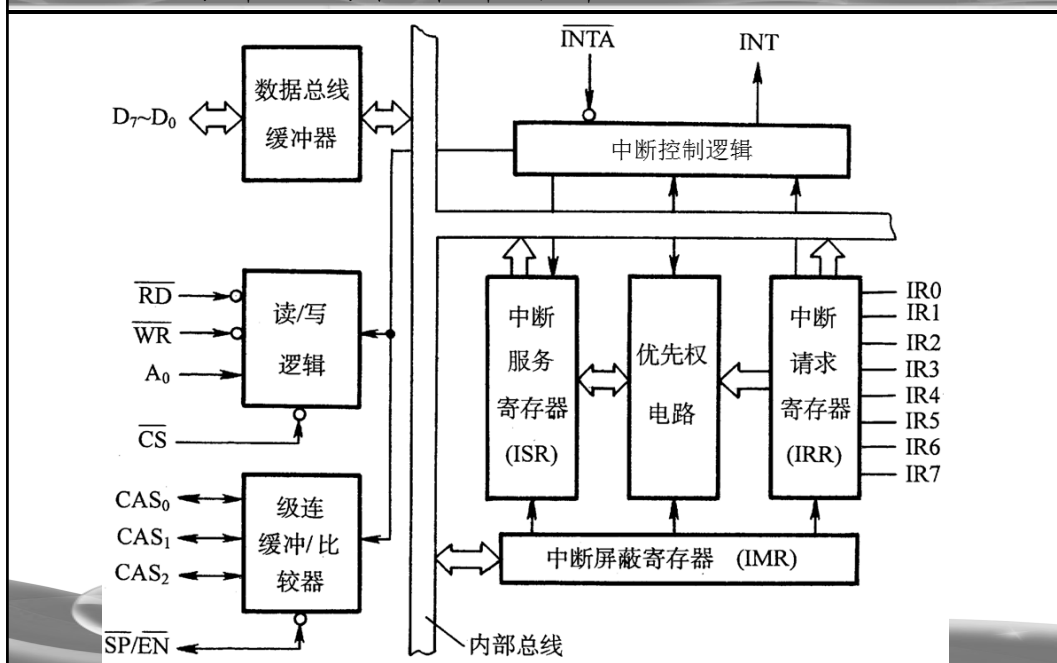
■ 面向外设的引脚

◆中断请求输入IR7~IR0（优先级：IR0最高）

8088中断系统（8259A实现）

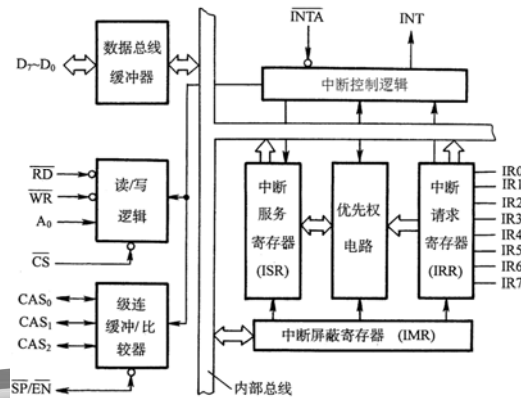


8259A内部结构和外部引脚



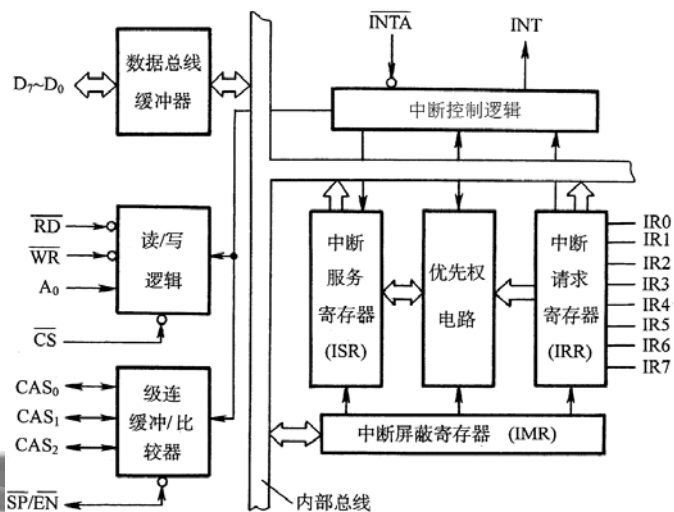
8259A结构

- 1.数据总线缓冲器
- 2.读/写逻辑
- 3.中断请求寄存器IRR（Interrupt Request Register）
- 4.中断服务寄存器ISR（Interrupt Service Register）
- 5.中断屏蔽寄存器IMR（Interrupt Mask Register）
- 6.中断控制逻辑
- 7.级联缓冲/比较器
- 8.优先权电路



8259A结构

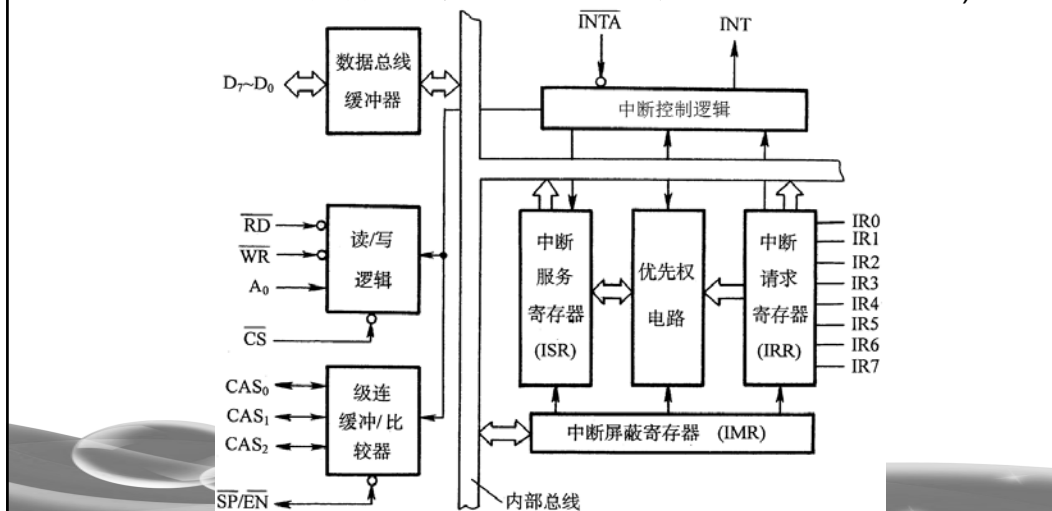
- ①数据总线缓冲器
 - 双向三态，与系统DB相连，传送控制字，状态，中断号等信息



8259A结构

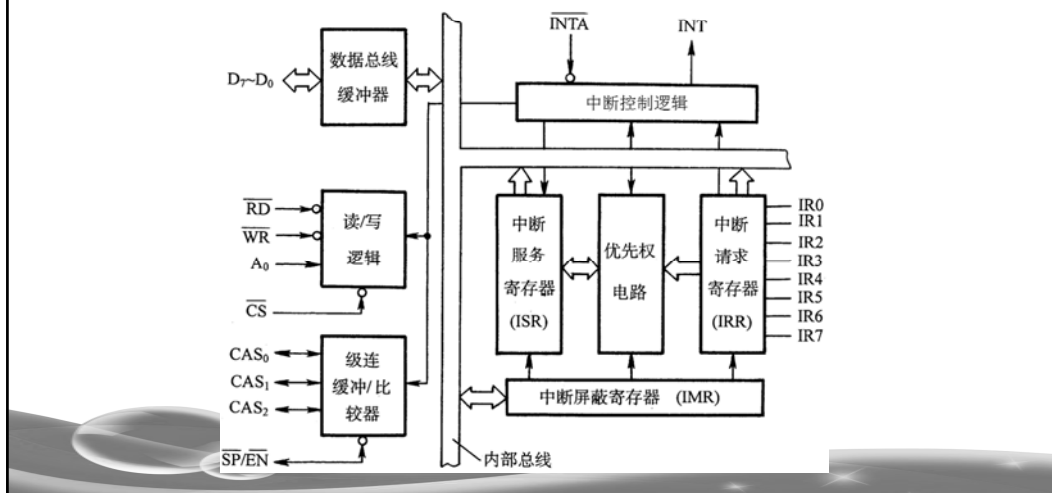
● ②读/写控制逻辑

- 控制片选 \overline{CS} ，地址线 A_0 ，以及 $\overline{WR}/\overline{RD}$ 线。执行包括写入命令（初始化和控制）或读取8259相应端口（IRR，ISR，IMR）。



● ③中断请求寄存器IRR

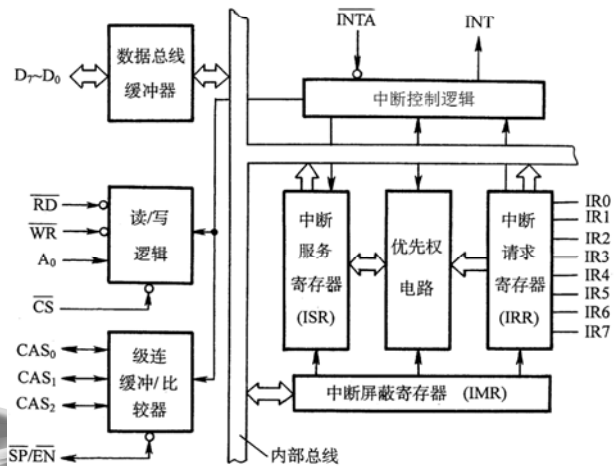
- 8位，寄存 $IR_0 \sim IR_7$ 是否有中断请求
- D_i 位表示 IR_i 脚有无中断请求：1有；0无



● ④中断服务寄存器ISR

■ 记录 $IR_0 \sim IR_7$ 中断源是否被服务中...

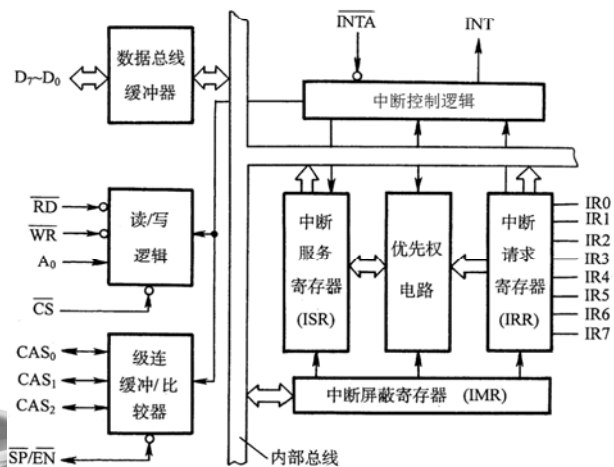
■ D_i 位表示 IR_i 是否正在进行 被服务：1是；0否



● ⑤中断屏蔽寄存器IMR

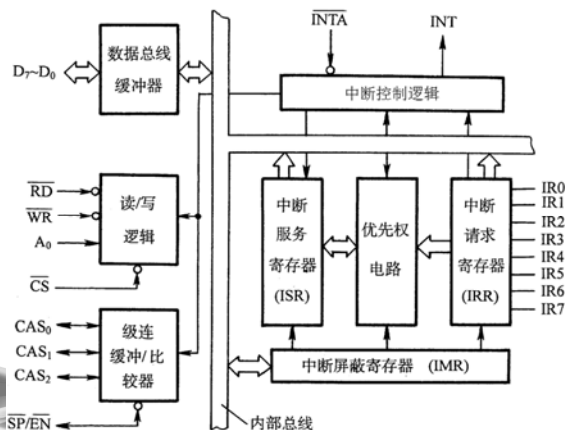
■ 8位，记录是否对 $IR_0 \sim IR_7$ 的中断申请进行屏蔽（禁止）

■ D_i 位表示 IR_i 中断申请是否进行屏蔽：1是；0否



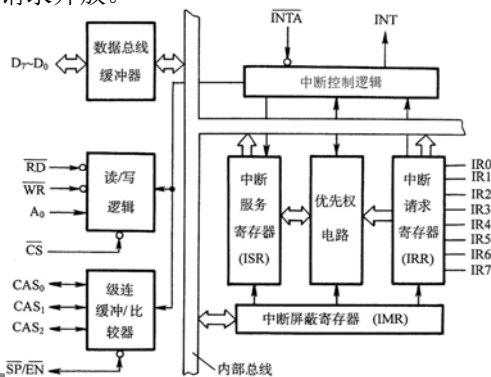
● ⑥中断控制逻辑

- 产生向CPU输出的中断请求信号INT
- 接收CPU送来的中断响应信号 \overline{INTA}
- 控制8259向DB上送出中断类型号N。

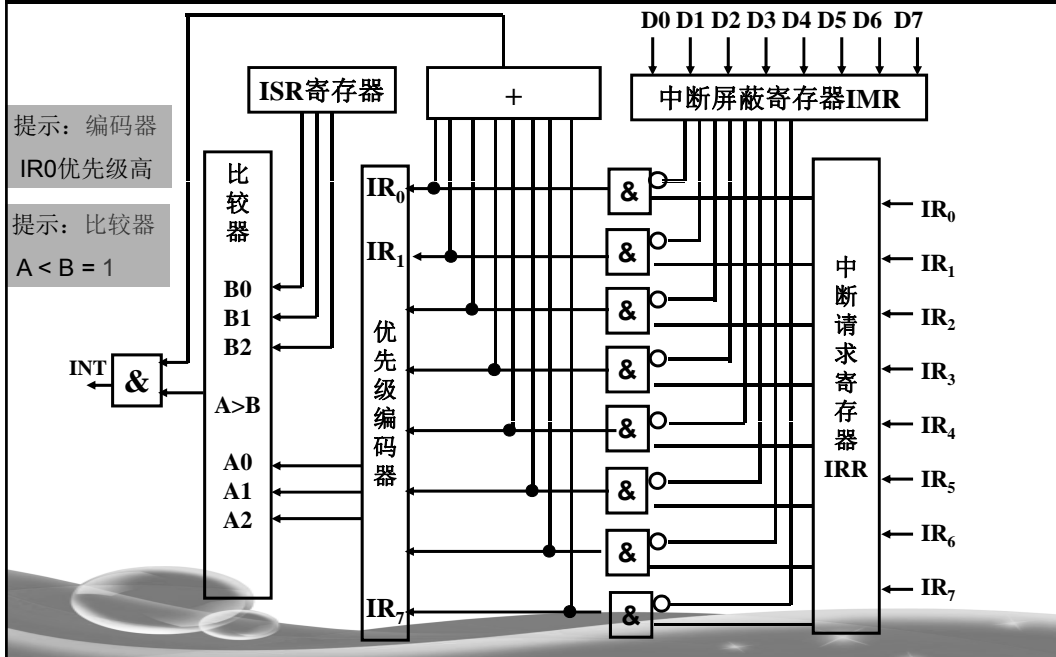


● ⑦ 优先权电路

- 选择IRR中优先权最高的中断源 IR_i ，并将其和ISR中的中断源比较，若 IR_i 优先级更高，则向CPU产生中断申请INT。
- 优先权的顺序为： $IR_0 > IR_1 > \dots > IR_7$
- 当某个中断源被服务时，它会禁止同级或低级中断请求的发生，但能向高一级的中断请求开放。



优先权电路（思考：如何阻止同级或低级中断源再次申请中断？）



优先权电路

● 工作原理

- 首先，由8个“与”门逻辑选出参加中断优先级排队中断请求级：由8位IRR与8位IMR分别送入“与”门输入端，只有当IRR位置“1”（有中断请求）且IMR位置“0”（开放中断请求）同时成立时，相应“与”门输出才为高电平，并送到优先级编码器的输入端参加编码。
- 其次，优先级编码器对参加排队的中断优先级进行编码，并从中选出当前最高优先级的代码（A2，A1，A0）。
- 最后，把ISR中当前正服务的优先级的代码（B2，B1，B0）与新来的中断请求的优先级代码（A2，A1，A0）一起送入比较器进行比较：当比较器输出有效且有中断请求时，与门将输出有效电平向CPU提出中断请求INT。
- 结论：当一个中断源被服务时，它会禁止同级或低级中断请求的发生，但能向高一级的中断请求开放。

● ⑧级联缓冲/比较器

■ $\overline{SP/EN}$: 从编程/缓冲器允许信号。

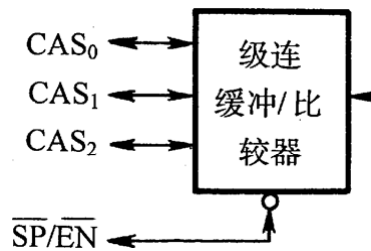
◆ 采用缓冲方式时（输出）：控制数据总线缓冲器的使能；

◆ 采用非缓冲方式（输入）：区分主从片。主片=1，从片=0。

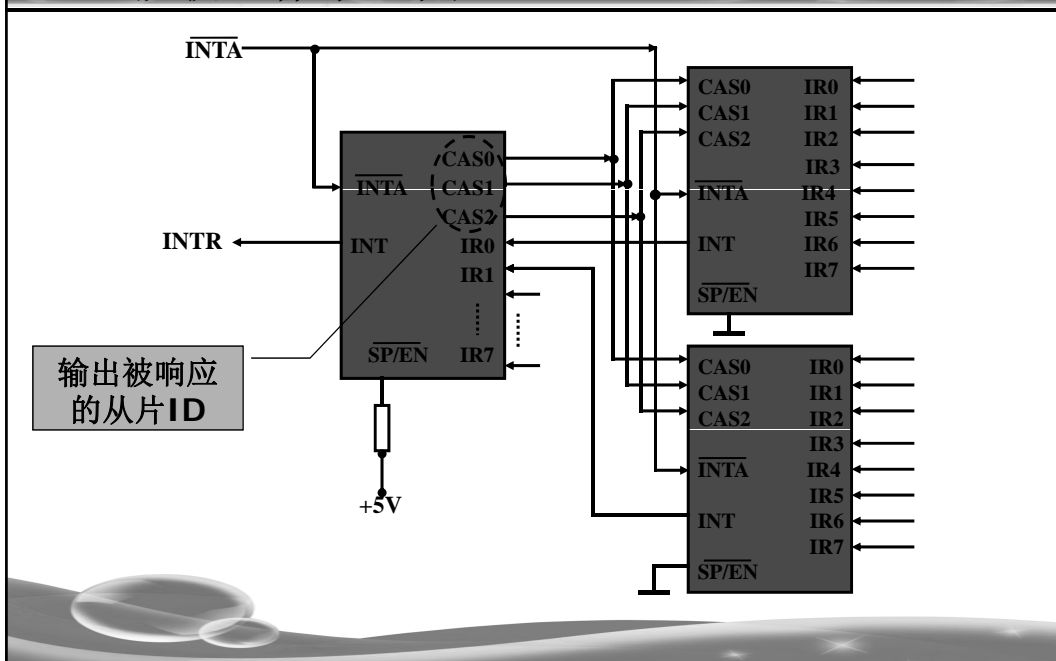
■ $CAS_{2\sim0}$: 级联信号线

◆ 主片：输出

◆ 从片：输入



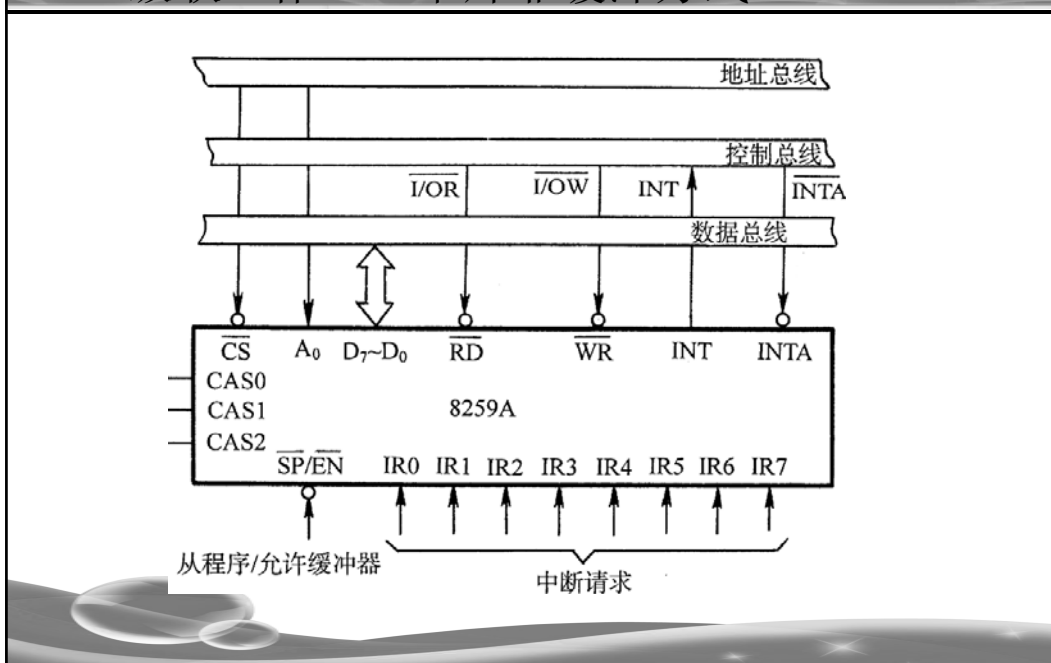
8259级联工作示意图



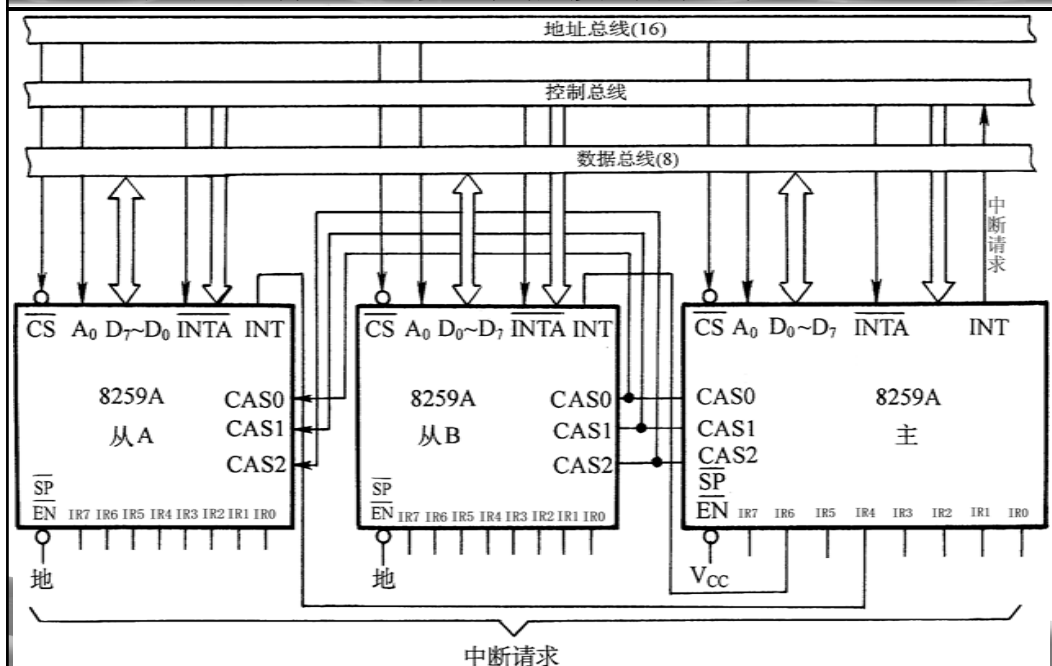
8259级联工作

- 8259A可以级联，1个主片最多可以级联8个从片
 - 级联时，主片CAS₀~CAS₂连至每个从片的CAS₀~CAS₂，输出被选中的从片ID，
 - 每个从片的中断请求信号INT，连至主片8259A某个中断请求输入端IR_i；主片的INT线连至CPU的中断请求输入端INTR
 - 主片在第1个响应周期内通过CAS₂~CAS₀送出从片ID，相应的从片在第2个响应周期内则将中断类型码N发送到数据总线上。

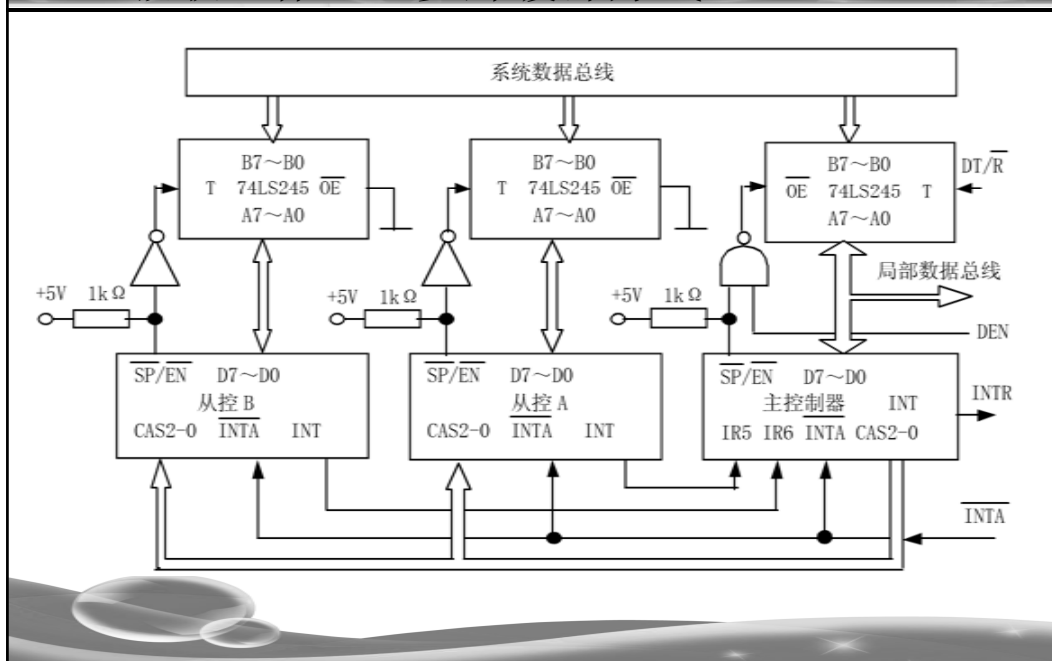
8259级联工作——单片非缓冲方式



8259级联工作——多片非缓冲方式



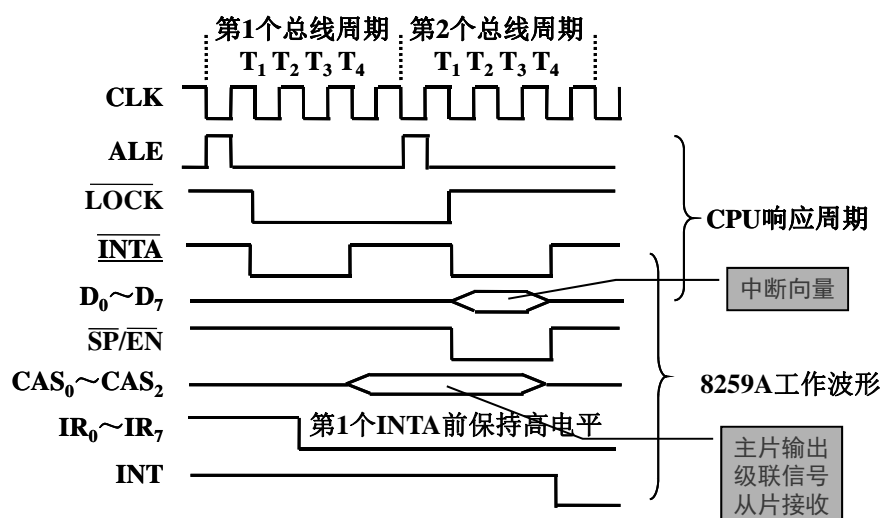
8259级联工作——多片缓冲方式



8259A的中断过程

- 1.当中断输入线（ IR_{7-0} ）有变高，中断请求寄存器IRR相应位置1
- 2.若中断请求线中至少有一位是中断允许的，则8259A通过INT引脚向CPU送出中断请求
- 3.若CPU开中断，则用 \overline{INTA} 响应。
- 4.8259A收第1个 \overline{INTA} 响应后
 - 使最高优先权的ISR位置位，相应IRR位复位。
 - ◆ 优先权的顺序为： $IR_0 > IR_1 \dots > IR_7$
 - 主片通过CAS局部总线输出从片ID。从片验证自己的ID。
- 5. 8259A收第2个 \overline{INTA} 响应后，（ID验证通过的）8259A向DB送出中断类型号N。
- 6.CPU读取DB上的N，自动计算中断向量的地址（ $N \times 4$ ），在AB上出现 $N \times 4$ ，读取其中的CS:IP并最终转向中断服务程序

8259A的中断过程



8259A的端口和操作

- 2个端口

- 按端口地址区分命令（偶地址A₀=0和奇地址A₀=1）
- 按顺序或特征位区分命令（同一端口地址）

\overline{CS}	\overline{WR}	\overline{RD}	A0	读写操作
0	0	1	0	写ICW1,OCW2,OCW3
0	0	1	1	写ICW2~ICW4,OCW1
0	1	0	0	读IRR,ISR,查询字
0	1	0	1	读IMR

第4节 8259A的工作方式

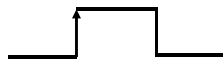
- 8259A的工作方式

- 1.引入中断请求（中断触发）的方式
- 2.连接系统总线的方式
- 3.级联方式
- 4.屏蔽中断源的方式
- 5.优先级排队的方式
- 6.结束中断的方式

- 1.引入中断请求（中断触发）的方式

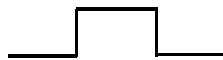
- ①边沿触发方式。

- ◆以正跳沿请求中断，维持高电平不会继续产生中断。



- ②电平触发方式。

- ◆以高电平申请中断，但响应中断后须及时清除高电平



- ③注意：查询方式。

- ◆CPU用软件查询确定中断源。外设通过8259申请中断，但不用INT向CPU申请中断。

- 2. 连接系统总线的方式

- 缓冲方式

- ◆ 通过数据总线缓冲器与CPU数据总线连接。

- ◆ $\overline{SP/EN}$ 用于启动数据总线缓冲器，不表示主/从关系。

- 非缓冲方式

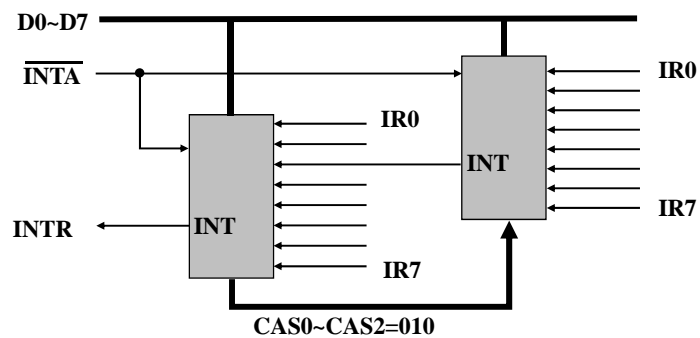
- ◆ 直接接至CPU数据总线。

- ◆ $\overline{SP/EN}$ 用于表示主/从芯片。

- 3. 级联方式

- 不级联——只用1片，如 PC/XT

- 级联——使用2~9片，如 PC/AT（2片）



● 4.屏蔽中断源的方式

■ ①通常屏蔽方式

◆利用操作命令字OCW1，设置屏蔽寄存器IMR

◆例：OCW1：1111 0011 开放IR3，IR2两个中断源。

■ ②特殊屏蔽方式

◆在某些场合，当执行某一个高优先级的中断服务程序时，若要求允许另一个优先级比它低的中断请求被响应，此时可采用特殊屏蔽方式。它可通过OCW3的D6D5=11来设定。

● 5.优先级排队的方式

■ ① 全嵌套方式。

◆优先级按0~7顺序排队，且只允许级别高的中断源去中断级别低的中断服务程序。常用缺省方式。

■ ② 自动轮换方式

◆中断服务结束后优先级降为最低（7），相邻的低优先级中断源自动升为最高，其余顺变。例：IR₂中断服务结束后：

IR0	IR1	IR2	IR3	IR4	IR5	IR6	IR7
5	6	7	0	1	2	3	4

◆每个中断源都有最高优先级的资格，故称“等优先级方式”

■ ③优先级指定轮换方式

■ ④ 特殊循环方式

- 6.结束中断的处理方式

- ①自动中断结束方式

- ◆在第二个INTA响应信号中，8259自动执行中断结束EOI指令，复位ISR中已置位的位。

- ◆EOI：中断结束字

- ②非自动中断结束方式

- ◆在中断服务程序返回之前，必须发中断结束EOI命令才能使ISR中的当前服务位清除。

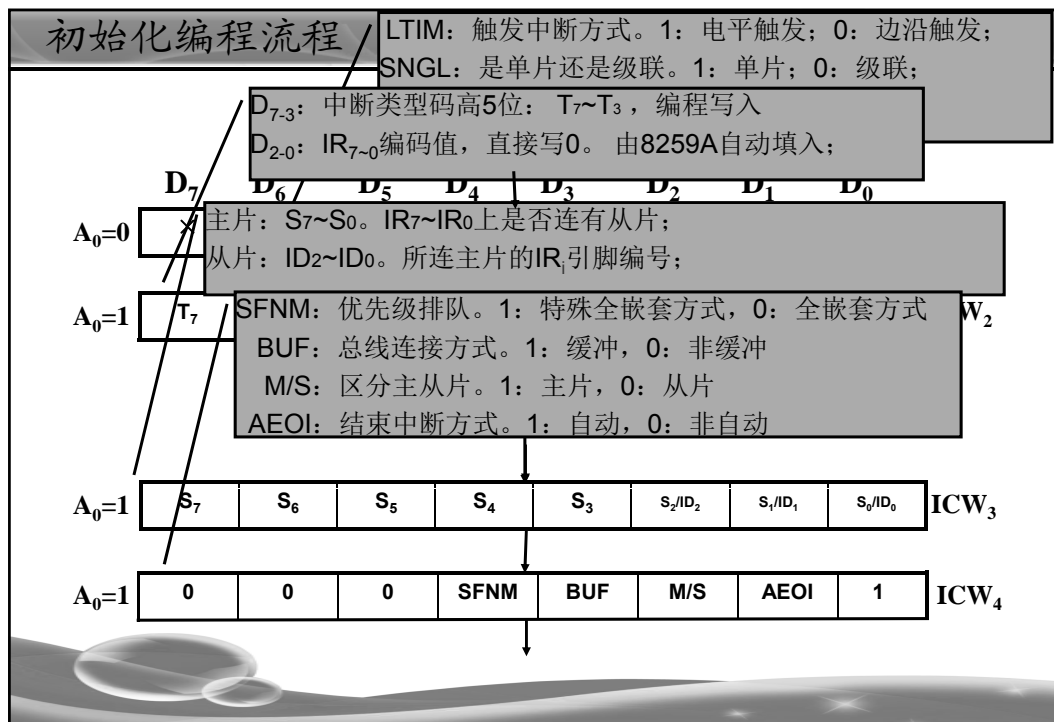
第5节 8259的编程和应用

8259的编程

- 8259的编程分为两个阶段
 - 初始化阶段
 - ◆ 在系统加电或复位后由初始化程序完成。
 - ◆ 设定工作方式、工作条件、中断类型码等。
 - ◆ 初始化命令 (Initialize Control Word)
 - ICW1 ~ ICW4
 - 操作控制阶段
 - ◆ 对8259的状态、中断方式和工作过程的控制。
 - ◆ 操作命令 (Operation Control Word)
 - OCW1 ~ OCW3

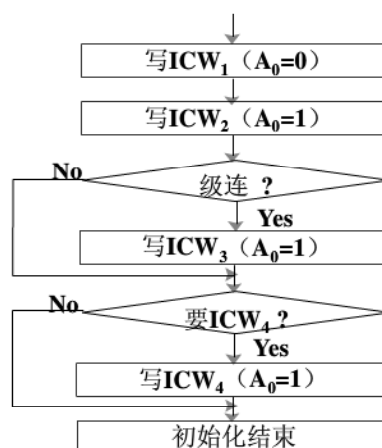
初始化命令 (ICW: Initialize Control Word)

- ICW1
 - 设置中断请求触发方式及芯片数目, 使8259A复位
- ICW2
 - 设置中断类型号N: 8位
- ICW3
 - 设置主从片的硬件连接方式
- ICW4
 - 设置优先级嵌套方式, 中断结束方式, 缓冲方式, 主从片



补充说明

- ICW的识别
 - 只要命令字D₄位为“1”，地址位A₀为“0”，就是ICW₁。
 - 接下来1~3字节就是ICW₂~ICW₄。
- 在不同的初始化要求中，ICW₂~ICW₄并非都必须使用。



补充说明

- ICW1复位芯片
 - ICW1清除IMR，缺省设置完全嵌套方式，IRQ_{0~7}优先级递降。
- ICW2设置中断类型号N（8位）
 - N高5位：初始编程输入。同片8259A的中断源N高5位相同。
 - N低3位：根据IR_i编码
 - ◆ 如IR₀的编码为000
 - ◆ 如IR₄的编码为100
 - ◆ 如IR₇的编码为111

初始化的例子

- 例：一片8259A，边沿触发方式；中断类型码为08H~0FH；
 - 用全嵌套、缓冲、非自动结束中断方式；
 - 8259A的端口地址为20H和21H。

```
MOV AL, 13H ;ICW1: 边沿触发, 单片, 设置IC4
OUT 20H, AL
```

```
MOV AL, 8 ;ICW2: 中断类型码为08~0FH
OUT 21H, AL
```

```
MOV AL, 0DH ;ICW4: 全嵌套、缓冲、非自动结束中断方式
OUT 21H, AL
```

初始化的例子: ICW1

- 例：一片8259A，边沿触发方式；中断类型码为08H~0FH；
 - 用全嵌套、缓冲、非自动结束中断方式；
 - 8259A的端口地址为20H和21H。

MOV AL , 13H ; ICW1: 边沿触发，单片，设置IC4
OUT 20H ,AL

	D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	
A ₀ =0	×	×	×	1	LTIM	×	SNGL	IC4	ICW ₁
	0	0	0	1	0	0	1	1	

初始化的例子: ICW2

- 例：一片8259A，边沿触发方式；中断类型码为08H~0FH；
 - 用全嵌套、缓冲、非自动结束中断方式；
 - 8259A的端口地址为20H和21H。

MOV AL , 8 ; ICW2: 中断类型码为08~0FH
OUT 21H ,AL

	T ₇	T ₆	T ₅	T ₄	T ₃	×	×	×	
A ₀ =1	0	0	0	0	1	0	0	0	ICW ₂

**思考：如果中断类型码为 020~027H，
如何设置ICW2 ？**

初始化的例子: ICW4

- 例: 一片8259A, 边沿触发方式; 中断类型码为08H~0FH;
 - 用全嵌套、缓冲、非自动结束中断方式;
 - 8259A的端口地址为20H和21H。

MOV AL,0DH ;ICW4: 全嵌套、缓冲、非自动结束中断
OUT 21H,AL

A ₀ =1	0	0	0	SFNM	BUF	M/S	AEOI	1	ICW ₄
	0	0	0	0	1	1	0	1	

- 8259的操作命令 (OCW: Operation Control Word)
 - 在初始后执行, 可不按顺序进行
 - 中断屏蔽操作命令 (OCW1)
 - 优先级选择/结束操作命令 (OCW2)
 - 中断查询操作 (OCW3)

◆操作命令字OCW₁——中断屏蔽/允许字
设置中断源IR_i的中断屏蔽/允许。

•例子：编程：屏蔽IR₀,IR₁,IR₇三个中断源

a) MOV AL, 83H	； 屏蔽IR ₀ , IR ₁ , IR ₇ , 开放IR ₂ ~IR ₆
OUT 21H, AL	； 写OCW ₁ （屏蔽字）

b) IN AL, 21H	； 读IMR（总是有效）
OR AL, 83H	； 屏蔽IR ₀ , IR ₁ , IR ₇ , 保留其他的
(或 AND AL, 83H)	； 开放IR ₂ ~IR ₆ , 保留其他的
OUT 21H, AL	； 写OCW ₁ （屏蔽字）

◆操作命令字OCW₂——中断方式字

D₇: 为1, 循环优先级方式；为0, 固定优先级方式。

D₆: 为1, D₂~D₀(L₂~L₀)有效；为0, D₂~D₀(L₂~L₀)无效。

D₅: 为1, EOI命令；为0, 非EOI命令。

D₂~D₀: ISR中的位编码。

•例子：中断服务程序中发出EOI指令

MOV AL, 20H	； 设置OCW ₂ 的EOI命令字
OUT 20H, AL	； 写OCW ₂
IRET	； 必须在中断结束前设置EOI

◆操作命令字OCW₃——状态操作字

D₆: 为1, 允许特殊屏蔽; 为0, 禁止特殊屏蔽。

D₅: 为1, 特殊屏蔽方式; 为0, 恢复原来优先级方式。

D₂: 为1, 中断查询方式; 为0, 非查询方式。

D₁: 为1, 允许读内部寄存器; 为0, 禁止读内部寄存器。

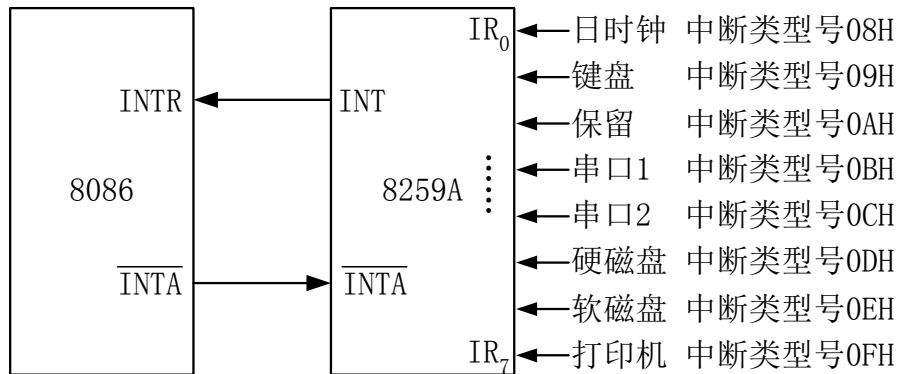
D₀: 为1, 读ISR; 为0, 读IRR。

● ICW/OCW操作功能表

类型	$\overline{\text{CS}}$	$\overline{\text{WR}}$	$\overline{\text{RD}}$	A ₀	功能	特征标志或流程
写命令字	0	0	1	0	数据总线→ICW ₁	ICW ₁ 的D ₄ 为1
	0	0	1	0	数据总线→OCW ₂	OCW ₂ 的D ₄ D ₃ 为00
	0	0	1	0	数据总线→OCW ₃	OCW ₃ 的D ₄ D ₃ 为01
	0	0	1	1	数据总线→OCW ₁ (屏蔽字)	无
	0	0	1	1	数据总线→ICW ₂ ~ICW ₄	ICW设置流程
读状态	0	1	0	0	IRR→数据总线	OCW ₃ 的D ₂ D ₁ D ₀ 为010
	0	1	0	0	ISR→数据总线	OCW ₃ 的D ₂ D ₁ D ₀ 为011
	0	1	0	0	中断查询字→数据总线	OCW ₃ 的D ₂ D ₁ D ₀ 为100
	0	1	0	1	IMR(屏蔽字)→数据总线	无

8259A在IBM-PC/XT中的应用

- 单片，端口地址：20H和21H
- $N = 0x8-0xF$
- 边沿触发，非自动中断结束方式，完全嵌套方式，缓冲方式



- 初始化编程

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	
×	×	×	1	LTIM	×	SNGL	IC4	ICW ₁
T ₇	T ₆	T ₅	T ₄	T ₃	×	×	×	ICW ₂
S ₇	S ₆	S ₅	S ₄	S ₃	S ₂ /ID ₂	S ₁ /ID ₁	S ₀ /ID ₀	ICW ₃
0	0	0	SFNM	BUF	M/S	AEOL	1	ICW ₄

;ICW₁，边沿触发，单片8259A，需ICW₄

```
MOV AL ,00010011B
```

```
OUT 20H ,AL
```

;设置ICW₂，中断类型号高5位为00001

```
MOV AL ,00001000B
```

```
OUT 21H ,AL
```

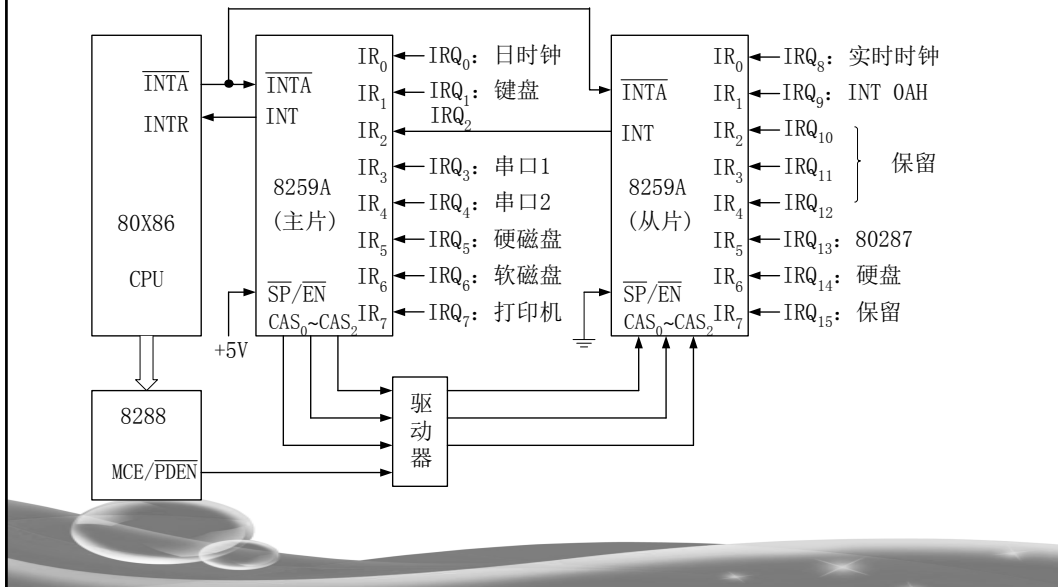
;设置ICW₄，非自动中断结束方式，完全嵌套方式，缓冲方式

```
MOV AL ,00001101B
```

```
OUT 21H ,AL
```

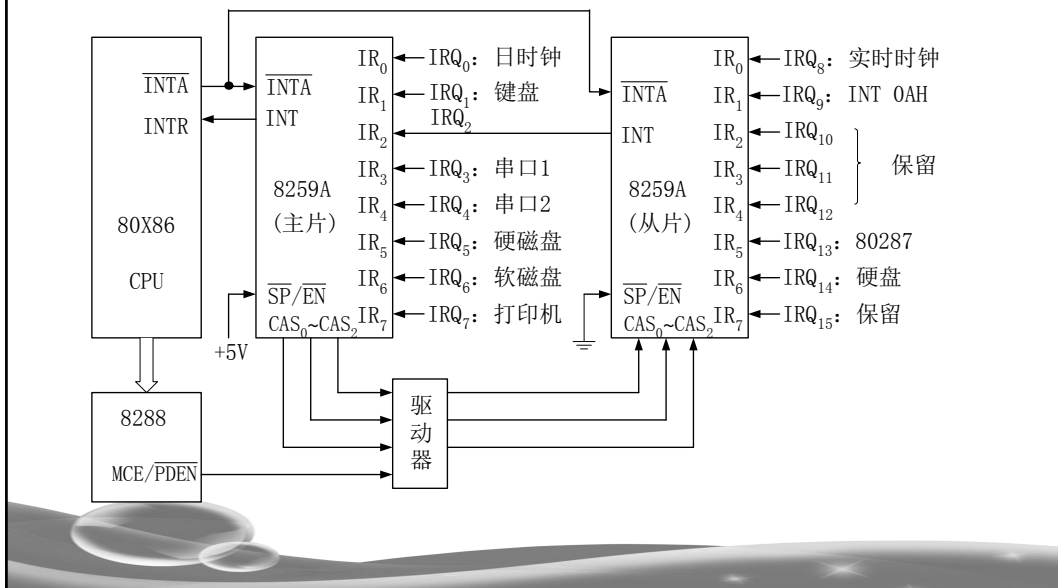
8259A在IBM-PC/AT的应用

- 两片8259A，地址：20H, 21H ; A0H, A1H



例子：写出两片8259A的初始化程序

- 两片8259A，地址：20H, 21H ; A0H, A1H



● 主片8259A

MOV AL, _____; 设置ICW1, 边沿触发, 需ICW4

OUT _____, AL

MOV AL, _____; 设置ICW2, 中断类型号的高5位为00001

OUT _____, AL

MOV AL, _____; 设置ICW3, 从片连到主片的IR₂上

OUT _____, AL

MOV AL, _____; 设置ICW4, 非缓冲, 非AEIOI, 特殊全嵌套方式

OUT _____, AL

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	
×	×	×	1	LTIM	×	SNGL	IC4	ICW ₁
T ₇	T ₆	T ₅	T ₄	T ₃	×	×	×	ICW ₂
S ₇	S ₆	S ₅	S ₄	S ₃	S ₂ /ID ₂	S ₁ /ID ₁	S ₀ /ID ₀	ICW ₃
0	0	0	SFNM	BUF	M/S	AEIOI	1	ICW ₄

● 从片8259A

MOV AL, _____; 设置ICW1, 边沿触发, 需ICW4

OUT _____, AL

MOV AL, _____; 设置ICW2, 中断类型号的高5位为01110

OUT _____, AL

MOV AL, _____; 设置ICW3, 设定从片级联于主片的IR₂

OUT _____, AL

MOV AL, _____; 设置ICW4, 非缓冲, 非AEIOI, 全嵌套方式

OUT _____, AL

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	
×	×	×	1	LTIM	×	SNGL	IC4	ICW ₁
T ₇	T ₆	T ₅	T ₄	T ₃	×	×	×	ICW ₂
S ₇	S ₆	S ₅	S ₄	S ₃	S ₂ /ID ₂	S ₁ /ID ₁	S ₀ /ID ₀	ICW ₃
0	0	0	SFNM	BUF	M/S	AEIOI	1	ICW ₄

80x86中断程序的编写过程

- 目的
 - 用自己的中断服务程序替换系统的中断服务程序
- 过程
 - 1.初始化向量表
 - 2.编写中断服务程序
 - 3.初始化8259A
 - 4.使能CPU的IF

1.初始化向量表

- 将用户自定义的中断服务程序入口地址放入向量表
- 例：将中断向量码为**48H**的服务程序入口地址放入向量表

```
MOV AX, 0000H
```

```
MOV DS, AX    ; 数据段从内存0地址开始（安排向量表）
```

```
MOV SI, 0120H ; 48H x 4
```

```
MOV BX, OFFSET INTService ; IP
```

```
MOV [SI], BX
```

```
MOV BX, SEG INTService ; CS
```

```
MOV [SI+2], BX
```

2.编写中断服务程序

```
DATA SEGMENT
    MESS DB 'This is a 8259A interrupt!', 0Ah, 0Dh, '$'
DATA ENDS

INTService PROC NEAR
    LEA DX, MESS    ; 显示字符串
    MOV AH, 09H
    INT 21H
    DEC BL
    MOV AL, 20H     ; 发送中断结束命令: EOI
    OUT 20H, AL
    IRET
INTService ENDP
```

3.初始化8259A

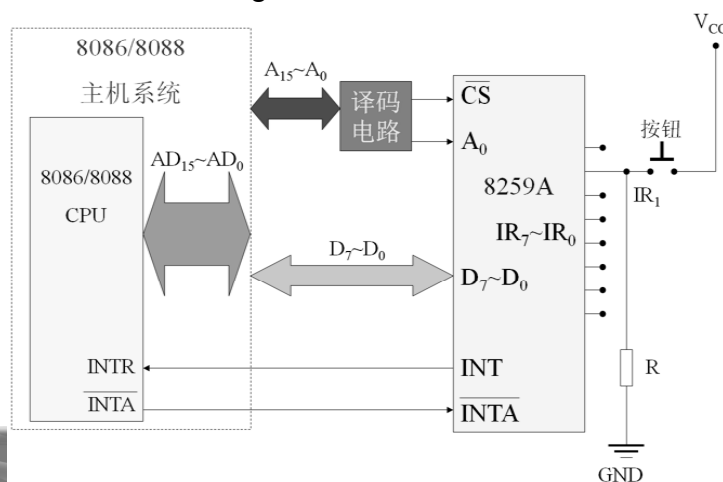
```
;ICW1, 边沿触发, 单片8259A, 需ICW4
MOV AL ,00010011B
OUT 20H ,AL
;设置ICW2, 中断类型号高5位为00001
MOV AL ,00001000B
OUT 21H ,AL
;设置ICW4, 非自动中断结束方式, 完全嵌套方式, 缓冲方式
MOV AL ,00001101B
OUT 21H ,AL
```

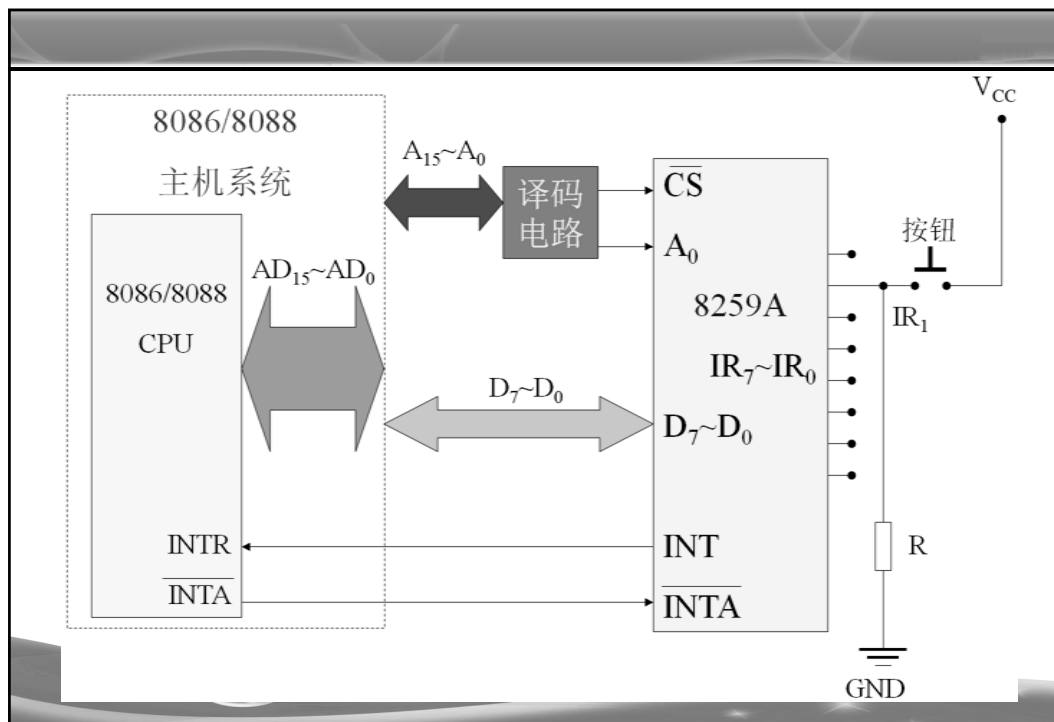
4.使能CPU的IF

STI

例子： 8259A应用

- 8088系统中8259A的端口地址是20H和21H，IR₁接单脉冲发生器，中断类型号是9，已初始化，编程使每接受到一个脉冲信号在屏幕上显示‘All Right!’，中断10次后退出程序。





课堂作业1: 8259A应用

编写一个程序段，将8259A（端口地址为50H，51H）中IRR，ISR，IMR的内容读出，存放到Buffer数据区。

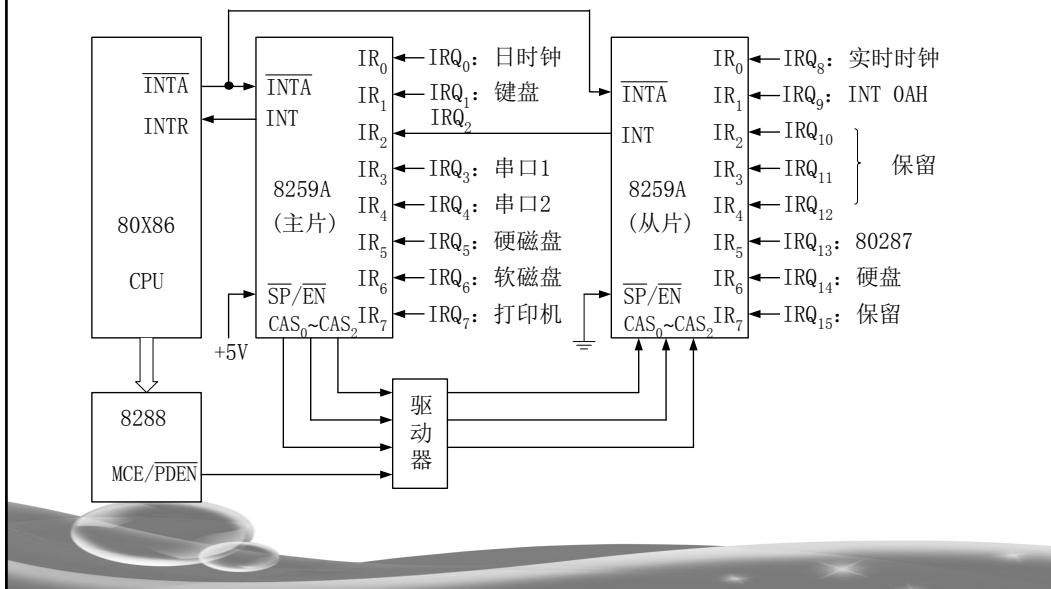
```

MOV AL, 00H
OUT 50H, AL ; 设置OCW3（读IRR命令）
IN AL, 51H ; 读IRR（中断请求寄存器）
MOV Buffer, AL
MOV AL, 00H
OUT 50H, AL ; 设置OCW3（读ISR命令）
IN AL, 51H ; 读ISR（中断服务寄存器）
MOV Buffer+1, AL
IN AL, 51H ; 读IMR（中断屏蔽寄存器）
MOV Buffer+2, AL

```

课堂作业2: 写出两片8259A的初始化程序

- 两片8259A, 地址: 20H, 21H ; A0H, A1H



主片8259A

MOV AL, _____; 设置ICW1, 边沿触发, 需ICW4

OUT _____, AL

MOV AL, _____; 设置ICW2, 中断类型号的高5位为00001

OUT _____, AL

MOV AL, _____; 设置ICW3, 从片连到主片的IR₂上

OUT _____, AL

MOV AL, _____; 设置ICW4, 非缓冲, 非AEIOI, 特殊全嵌套方式

OUT _____, AL

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	
×	×	×	1	LTIM	×	SNGL	IC4	ICW ₁
T ₇	T ₆	T ₅	T ₄	T ₃	×	×	×	ICW ₂
S ₇	S ₆	S ₅	S ₄	S ₃	S ₂ /ID ₂	S ₁ /ID ₁	S ₀ /ID ₀	ICW ₃
0	0	0	SFNM	BUF	M/S	AEIOI	1	ICW ₄

● 从片8259A

MOV AL, _____ ; 设置ICW1, 边沿触发, 需ICW4

OUT _____, AL

MOV AL, _____ ; 设置ICW2, 中断类型号的高5位为01110

OUT _____, AL

MOV AL, _____ ; 设置ICW3, 设定从片级联于主片的IR₂

OUT _____, AL

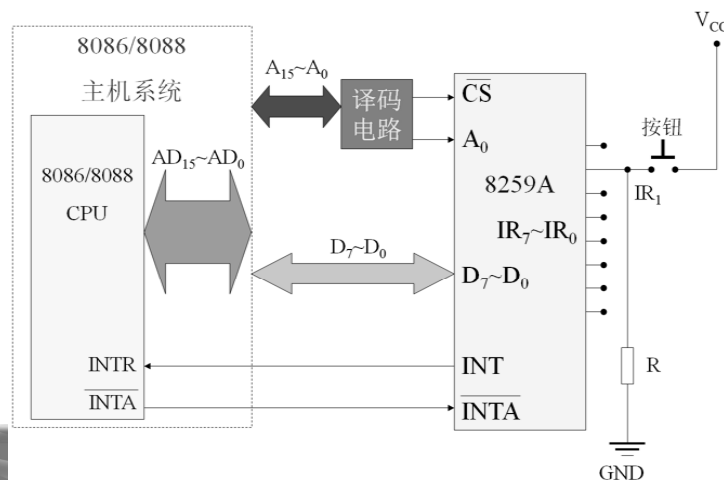
MOV AL, _____ ; 设置ICW4, 非缓冲, 非AEOI, 全嵌套方式

OUT _____, AL

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	
×	×	×	1	LTIM	×	SNGL	IC4	ICW ₁
T ₇	T ₆	T ₅	T ₄	T ₃	×	×	×	ICW ₂
S ₇	S ₆	S ₅	S ₄	S ₃	S ₂ /ID ₂	S ₁ /ID ₁	S ₀ /ID ₀	ICW ₃
0	0	0	SFNM	BUF	M/S	AEOI	1	ICW ₄

课堂作业3: 8259A应用

- 8088系统中8259A的端口地址是20H和21H, IR₁接单脉冲发生器, 中断类型号是9, 已初始化, 编程使每接受到一个脉冲信号在屏幕上显示 'All Right!', 中断10次后退出程序。



```

stack1 segment stack
    dw 20h dup(0)
stack1 ends
code segment
charbf        'All Right',0Ah,0Dh,'$' ;显示字符串
    assume cs:code,ss:stack1
start:mov dx,                  
    mov ds,dx
    mov dx,                  
    mov al,       ;中断类型号
    mov ah,      
                  ;装中断向量表
    mov       ,       ;循环10次

```

```

    mov al,      
    out       ,       ;OCW1, 置IMR=1111,1101B
           ;开中断
next:hlt ;等待外部中断
    loop       
    mov ah,       ;程序结束, 返回DOS
                 
charint:                     ;中断服务程序入口
    mov ah,      
    int 21h ;显示字符串
    mov al,       ; OCW2=0010,0000B
                         ; OCW2, 发中断结束命令
          

```