

接口与通讯实验 报告 3

Hollow Man

一、实验环境

一台带有装有 Windows XP 系统的实验室计算机，一台实验箱。

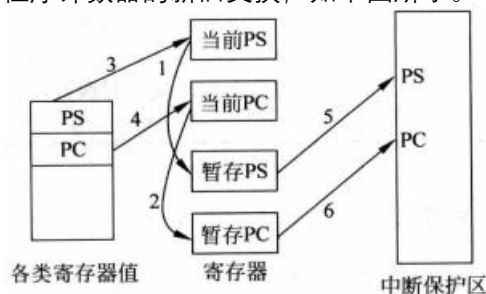
二、实验目的

(1) 掌握 8259A 中断控制器工作原理，熟悉实验中涉及 8259A 中的中断屏蔽寄存器 IMR 和中断服务寄存器 ISR 等的用法。

(2) 掌握外部中断源的引入,学会中断处理程序的编写。

(3) 掌握可屏蔽中断和不可屏蔽中断两种中断。

(4) 了解中断过程：进入中断、保存现场信息、判断中断源、中断处理、恢复现场退出中断及 PS 程序状态，PC 程序计数器的新旧交换，如下图所示。



说明：

(1) 进入中断时，保存当前的 PS 和 PC 值，即保护现场。

(2) 判断中断源、中断处理。

(3) 恢复现场退出中断及 PS 程序状态，PC 程序计数器的新旧交换。

三、芯片介绍

1. 中断

首先，中断分硬件中断和软件中断两种。中断为计算机的硬件设备和软件部件提供了一种相互交流的途径，这就是它的作用。那么，都有哪些中断呢？它们又是怎样实现这种交流的呢？

PC 中的 CPU 通常都是 Intel 80x86 处理器，它由多条引脚用来中断 CPU 的当前工作，并使它转去进行其他工作。每条中断引脚上都连接着一些硬件设备（如定时器），其作用是为这条引脚提供一个特定的电压。当中断事件发生时，处理器会停止执行当前正在执行的软件，保存当前的操作状态，然后去“处理”中断。处理器中事先已经装有一张中断向量表（如表 6.3 所示），其中列出了每个中断号以及当某个特定中断发生时所应执行的程序。

常见的中断:

中断(十六进制)	描 述
5	屏幕打印服务
10	视频显示服务(MDA、CGA、EGA、VGA)
11	获得设备清单
12	获得内存大小
13	磁盘服务
14	串行口服务
15	杂项功能服务
16	键盘服务
17	打印机服务
1A	时钟服务
21	DOS 函数
2F	DOS 多路共享服务
33	鼠标器服务
67	EMS 服务

2. 8259 芯片

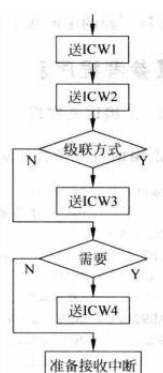
8259A 有两类编程命令: 初始化命令字 (ICW) 和操作命令字 (OCW)。8259A 初始化编程与中断向量的装入一样, 在 PC 中是由系统软件来做的, 并且开机上电就已经做好了, 不需要也不允许用户去做; 否则, 将对微机的中断系统产生很大的影响甚至是破坏。所以, 8259A 初始化, 一般在没有配置完善的操作系统的单板机微机上进行。

这里只是为了学习 8259A 初始化编程如何进行。如果是在 PC 上开发中断程序, 则不要使用 ICW1-ICW4 去初始化, 因为系统已经做好了。只需要使用 8259A 的两个操作命令字 OCW1 和 OCW2 进行中断的屏蔽和开放及发中断结束命令。在实际中, OCW3 用得很少。

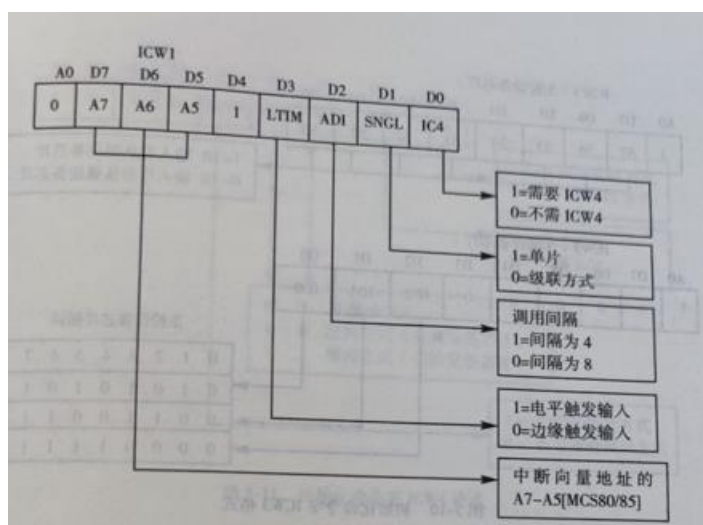
初始化命令字 ICW 的方法。在中断系统运行之前, 系统中的每一个 8259A 必须按先后次序接收 CPU 的 2~4 个 ICW 初始化命令字进行初始化。初始化顺序放在 8259A 之前, 作为主程序的一部分。初始化命令一定要按规定的顺序写入。

对于 16 位微机 ICW1 ICW2 和 ICW4 是必须有的, 而 ICW3 则需看是否多片使用, 若是, 则写 ICW3, 而且主从片的 ICW3 要分别写出。286 以上 PC 的中断系统有 15 级中断, 使用两片 8259A, 因此命令字 ICW1-ICW4 都要写出。操作命令字 OCW 的写法。当处理器对中断控制器完成初始化编程后, 8259A 就处于准备就绪状态, 等待接收外界的中断请求, 进行完全嵌套的中断管理。如果用户改变初始化设定的操作方式, 可以通过 CPU 发操作命令 OCW 对中断控制器进行动态控制。8259A 的 OCW 与 ICW 不同, 不需要按照顺序发送, 一般也不要求安排在程序开头, 而是根据需要在程序中任意安排。

8259A 初始化流程如下图所示:

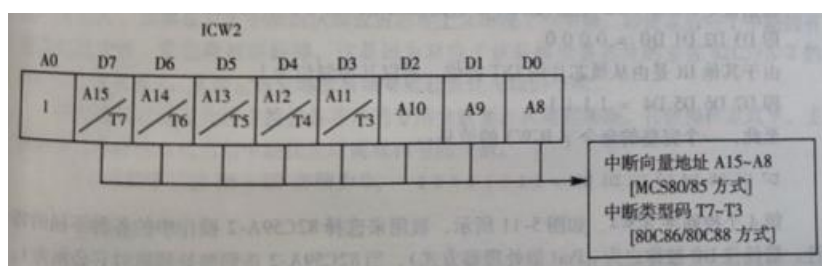


ICW1



ICW2

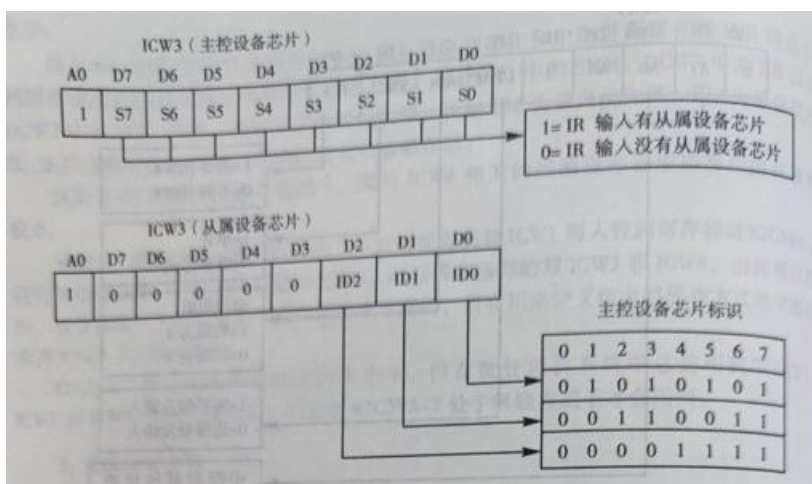
第 2 个初始化命令字 ICW2 在 80c86 微处理器系统中执行单一的功能，如下图所示，它的最左 5 位 (D7~D3) 定义了一个固定的二进制码 T7~T3,它是中断类型号的前 5 位。当 82c59A2 将相应有效输入中断类型号 (3 位) 送到总线上时，它自动与 T3~T7 结合形成 1 个 8 位中断类修号, ICW2 的最右 3 位未用。当该命令字送到总线上时 A0 输入必须为 1。



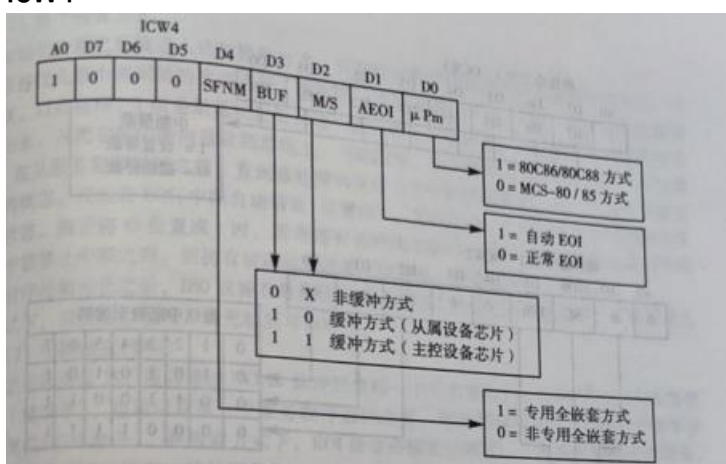
ICW3

如果把 82CS9A.2 配置成级联方式，就需要初始化命令字 ICW3。可用于不同的功能，这取决于该芯片是主控芯片还是从属芯片。在 82C59A.2 芯片，位 D0~D7 标上 S0~S7。这些位分别对应于 IR0~IR7 的输入。它们标识了相应的 IR 是由从属芯片的 INT 输出提供，还是直接由外部电路提供。为 1,则表示相应的 IR 输入是由从属芯片提供。

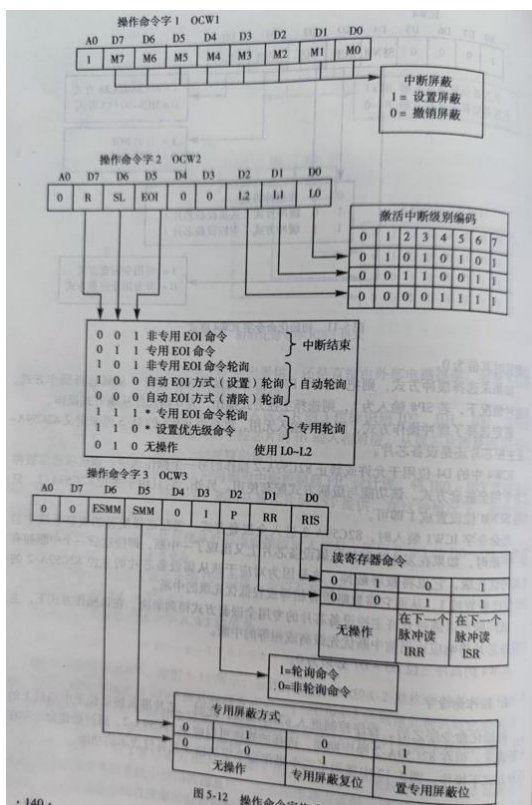
另一方面,从属芯片的 ICW3 用来装入该芯片的 3 位标识码 ID2、ID1、ID0,必须与从属芯片的 INT 输出相连的主控芯片的 1R 输入相对应。从属芯片需要比较主控芯片在 CAS0 - CAS2 上的级联码输出。



ICW4



OCW



三、实验内容

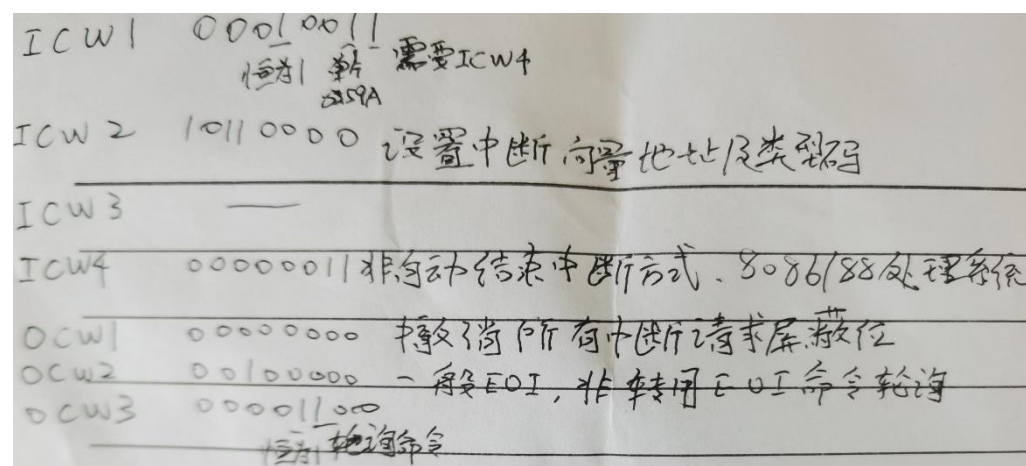
实验一

通过开关控制触发中断

四、实验原理

实验一

实验控制字



实验代码

```

DATA        SEGMENT
I8259_1     EQU    2B0H        ;8259 的 ICW1 端口地址
I8259_2     EQU    2B1H        ;8259 的 ICW2 端口地址
I8259_3     EQU    2B1H        ;8259 的 ICW3 端口地址
I8259_4     EQU    2B1H        ;8259 的 ICW4 端口地址
O8259_1     EQU    2B1H        ;8259 的 OCW1 端口地址
O8259_2     EQU    2B0H        ;8259 的 OCW2 端口地址
O8259_3     EQU    2B0H        ;8259 的 OCW3 端口地址

MES1        DB 'YOU CAN PLAY A KEY ON THE KEYBOARD! ',0DH,0AH,24H
MES2        DD    MES1

MESS1       DB 'HELLO! THIS IS INTERRUPT * 0 *!',0DH,0AH,'$'
MESS2       DB 'HELLO! THIS IS INTERRUPT * 1 *!',0DH,0AH,'$'
MESS3       DB 'HELLO! THIS IS INTERRUPT * 2 *!',0DH,0AH,'$'
MESS4       DB 'HELLO! THIS IS INTERRUPT * 3 *!',0DH,0AH,'$'
MESS5       DB 'HELLO! THIS IS INTERRUPT * 4 *!',0DH,0AH,'$'
MESS6       DB 'HELLO! THIS IS INTERRUPT * 5 *!',0DH,0AH,'$'
MESS7       DB 'HELLO! THIS IS INTERRUPT * 6 *!',0DH,0AH,'$'
MESS8       DB 'HELLO! THIS IS INTERRUPT * 7 *!',0DH,0AH,'$'

DATA        ENDS

STACKS      SEGMENT
            DB 100 DUP(?)

STACKS      ENDS

STACK1      SEGMENT STACK

```

```

        DW 256 DUP(?)
STACK1  ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA,SS:STACKS,ES:DATA

.386
START:  MOV     AX,DATA
        MOV     DS,AX
        MOV     ES,AX
        MOV     AX,STACKS
        MOV     SS,AX
        MOV     DX,I8259_1      ;初始化 8259 的 ICW1
        MOV     AL,13H          ;边沿触发、单片 8259、需要 ICW4
        OUT     DX,AL
        MOV     DX,I8259_2      ;初始化 8259 的 ICW4
        MOV     AL,0B0H          ;非自动结束 EOI
        OUT     DX,AL
        MOV     AL,03H
        OUT     DX,AL
        MOV     DX,O8259_1      ;初始化 8259 的 OCW1
        MOV     AL,00H          ;打开 IR0 和 IR1 的屏蔽位
        OUT     DX,AL
QUERY:  MOV     AH,1            ;判断是否有按键按下
        INT     16H
        JNZ     QUIT            ;有按键则退出
        MOV     DX,O8259_3      ;向 8259 的 OCW3 发送查询命令
        MOV     AL,0CH
        OUT     DX,AL
        IN      AL,DX           ;读出查询字
        MOV     AH,AL
        AND     AL,80H
        TEST    AL,80H          ;判断中断是否已响应
        JZ      QUERY          ;没有响应则继续查询
        MOV     AL,AH
        AND     AL,07H
        CMP     AL,00H
        JE      IRQISR          ;若为 IR0 请求，跳到 IR0 处理程序
        CMP     AL,01H
        JE      IRQISR          ;若为 IR1 请求，跳到 IR1 处理程序
        CMP     AL,02H
        JE      IRQISR
        CMP     AL,03H
        JE      IRQISR
        CMP     AL,04H
        JE      IRQISR

```

```

        CMP     AL,05H
        JE      IR5ISR
        CMP     AL,06H
        JE      IR6ISR
        CMP     AL,07H
        JE      IR7ISR
        JMP     QUERY
IR0ISR: MOV     AX,DATA
        MOV     DS,AX
        MOV     DX,OFFSET MESS1      ;显示提示信息
        MOV     AH,09
        INT     21H
        JMP     EOI
IR1ISR: MOV     AX,DATA
        MOV     DS,AX
        MOV     DX,OFFSET MESS2      ;显示提示信息
        MOV     AH,09
        INT     21H
        JMP     EOI
IR2ISR: MOV     AX,DATA
        MOV     DS,AX
        MOV     DX,OFFSET MESS3      ;显示提示信息
        MOV     AH,09
        INT     21H
        JMP     EOI
IR3ISR: MOV     AX,DATA
        MOV     DS,AX
        MOV     DX,OFFSET MESS4      ;显示提示信息
        MOV     AH,09
        INT     21H
        JMP     EOI
IR4ISR: MOV     AX,DATA
        MOV     DS,AX
        MOV     DX,OFFSET MESS5      ;显示提示信息
        MOV     AH,09
        INT     21H
        JMP     EOI
IR5ISR: MOV     AX,DATA
        MOV     DS,AX
        MOV     DX,OFFSET MESS6      ;显示提示信息
        MOV     AH,09
        INT     21H
        JMP     EOI
IR6ISR: MOV     AX,DATA

```

```
        MOV    DS,AX
        MOV    DX,OFFSET MESS7      ;显示提示信息
        MOV    AH,09
        INT    21H
        JMP    EOI
IR7ISR:  MOV    AX,DATA
        MOV    DS,AX
        MOV    DX,OFFSET MESS8      ;显示提示信息
        MOV    AH,09
        INT    21H
EOI:     MOV    DX,O8259_2           ;向 8259 发送中断结束命令
        MOV    AL,20H
        OUT    DX,AL
        JMP    QUERY
QUIT:    MOV    AX,4C00H             ;结束程序退出
        INT    21H
CODE     ENDS
        END    START
```