

## 实验十 键盘中断

### 一、实验目的

1. 了解 Intel 8086CPU 的中断处理功能以及 IBM-PC 的中断结构。
2. 了解 8259 中断控制器的使用。
3. 掌握键盘中断的编程，观察中断的执行情况。

### 二、实验任务

#### 1. 基本功能：

要求每按下任意一个键就向 CPU 发出中断请求信号，该信号由 8259 的 IRQ1 引入，中断类型为 09，CPU 响应中断后转入执行 KEYINTS 中断服务程序，并在屏幕上显示“OK!”，按下 10 次键后返回 DOS。

#### 2. 附加功能：

- 1) 通过 DOS 系统功能调用的 25H、35H 功能实现中断向量的设置和读取；

- 2) 改变按键后屏幕显示的字符串内容和返回 DOS 之前的按键次数，

比如：按键后在屏幕上显示“KEYINT”，按下 9 次键后返回 DOS。

按键显示的字符串内容和返回 DOS 之前的按键次数 各位同学可以自己设置，尽量不要太雷同，

要求显示字符串的字符个数>3，按键次数>8。

- 3) 在按键后显示的字符串前面加上按键次数，在字符串后面加个空格，这样两次按键显示字符串之间有个空格间隔区分一下；

- 4) 按键次数达到后(比如 9 次)，不等 25 行太阳图标显示完，立即返回 DOS；

- 5) 修改显示字符的属性，如，红底白字，蓝底黄字，……；

- 6) 其他：鼓励同学们自己增加其他的附加功能。

### 三、实验原理

键盘与主机是通过 5 芯螺旋形的电缆相连的，其中包括数据线、时钟线、复位线、+5v 电源线和地线。(电缆插入系统板后部的插座)

每当有键按下或释放时，键盘以串行方式向系统板的键盘接口电路传送数据，即扫描码。一个扫描码移位传送完，键盘接口电路便向主机发出中断请求信号 IRQ1(中断类型码为 09H)，此信号送到 8259A 产生中断请求。

CPU 响应中断请求时，查中断向量表，从 09H×4 开始的连续四个单元中取出中断向量（IRQ1 中断服务程序 KEYINTS 的入口地址指针），转去执行中断服务程序 KEYINTS。

主程序和键盘中断服务程序的流程图如图 1-10-1 和图 1-10-2 所示。请根据流程图编写主程序和键盘中断服务程序。

在主程序中应先读取并保存中断类型 09H 的原中断向量，然后再设置新的中断向量，即将中断服务程序 KEYINTS 入口地址的偏移量和段基址存入以 09×4 为起始地址的四个单元内。

在键盘中断程序 KEYINTS 中，保护现场、开中断之后，就通过 8255A 的 PA 口(PA 口地址为 60H)读取键盘扫描码，接着从 8255A PB 口(PB 口地址为 61H)的 PB7 输出一个正脉冲(即 PB7 先输出高电平，再输出低电平)，先输出的高电平信号反相之后控制键盘状态触发器的清零端，使 IRQ1 清零，撤消中断请求信号。再输出的低电平信号允许位寄存器输出数据，这样就为传递下一个键盘扫描码作好了准备。

8255A PB 口的 PB7 输出正脉冲来发键盘状态复位命令的具体指令如下：

```
IN      AL, 61H      ; 输入 PB 口的当前值
OR      AL, 80H      ; PB7 置 1
OUT     61H, AL
AND     AL, 7FH      ; PB7 清零
OUT     61H, AL
```

在键盘中断程序结束前,应发出“中断结束”(EOI)命令(控制字为 20H),给 8259A 的操作命令字 OCW1(端口地址为 20H)。具体指令如下:

```
MOV AL, 20H
OUT 20H, AL
```

然后,实现中断返回。

另外,当键按下时,送向主机的扫描码是键编号,而键释放时,扫描码为键编号加 80H(即第 7 位置 1)。例如,按下和释放“A”键将向主机发送两个扫描码:1EH 和 9EH。

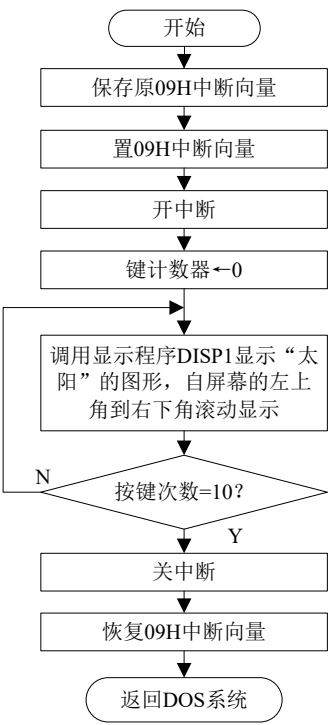


图 1-10-1

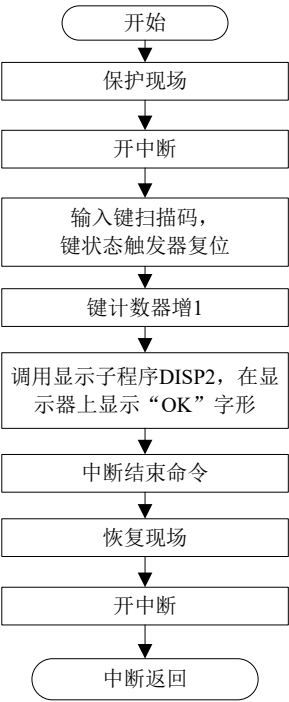


图 1-10-2

以下是主程序中显示“太阳”图形的子程序 DISP1,仅供参考。

```
DISP1 PROC FAR
    PUSH AX
    PUSH BX
    PUSH CX
    PUSH DX
    MOV AH, 15      ; 读当前显示状态
    INT 10H
    MOV AH, 0       ; 设置显示方式
    INT 10H
    MOV DX, 0       ; 行号为 0, 列号为 0
REPT: MOV AH, 2     ; 设置光标位置
    INT 10H
    MOV AL, 0FH     ; 0FH--太阳图形的 ASCII 码
    MOV CX, 1       ; 显示字符个数
```

```

        MOV    AH, 10        ; 写字符
        INT    10H
        CALL   DELAY
        SUB    AL, AL
        MOV    AH, 10        ; 清除原图形
        INT    10H
        INC    DH            ; 行号+1
        ADD    DL, 2         ; 列号+2
        CMP    DH, 25        ; 是否到 25 行?
        JB     REPT
        POP    DX
        POP    CX
        POP    BX
        POP    AX
        RET
DISP1   ENDP

```

以下是中断服务程序中显示“OK!”字符串子程序 DISP2，仅供参考。

```

DISP2   PROC   FAR
        PUSH   CX
        PUSH   BX
        PUSH   AX
        MOV    CX, 3
NEXTC:  LODSB                    ; 字符串“OK!”在数据段中定义，AL←[SI]
        MOV    AH, 0EH          ; 写字符，并移动光标
        MOV    BX, 01
        INT    10H
        CALL   DELAY
        LOOP   NEXTC
        POP    AX
        POP    BX
        POP    CX
        RET
DISP2   ENDP

```

以下是延时 1 秒子程序。

```

DELAY   PROC
        PUSH   CX
        PUSH   DX
        MOV    DX, 20
DL500:  MOV    CX, 2801
DL10ms: LOOP   DL10ms
        DEC    DX
        JNZ    DL500

```

```
    POP    DX
    POP    CX
    RET。
DELAY  ENDP
```

注：延时 1 秒子程序可以用实验九调试好的延时子程序，循环延时、调用 DOS 中断调用延时都可以。

#### 四、实验设备

IBM-PC / XT 微机 一台

#### 五、实验预习要求

1. 正确理解实验目的、内容和原理。
2. 根据流程图在实验前编写好主程序和键盘中断服务程序。

#### 六、实验报告要求

1. 整理出调试正确的主程序和中断服务程序，并加以注释。
2. 写出实验结果。
3. 分析在实验中所出现的问题。
4. 回答思考题。

#### 七、思考题

1. 键盘上某个键按下和释放时都会向 8259 发出中断请求，要求只在键按下时显示 ‘OK!’，键释放时不显示，在中断服务程序 KEYINTS 中是如何实现的？
2. 完成附加功能 4(按键次数达到后(比如 9 次)，不等 25 行太阳图标显示完，立即返回 DOS)的同学，请简要说明：
  - 1) 是如何实现该附加功能的？
  - 2) 如果程序运行结束后，键盘不能正常使用（按键没反应），可能是哪些原因？

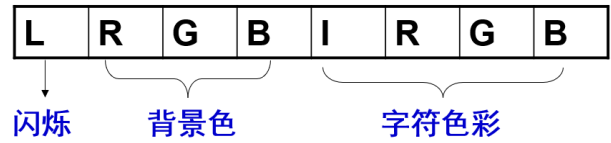
附：屏幕显示(INT 10H)

功 能	入 口 参 数	出 口 参 数
置显示模式	(AH)=0 (AL)=模式编码 字符模式 0—40 X 25 黑白 1—40 X 25 彩色 2—80 X 25 黑白 3—80 X 25 彩色 图形模式 4—320 X 200 彩色 5—320 X 200 黑白 6—640 X 200 黑白 10—640 X 350 EGA	无
设置光标大小	(AH)=1 (CH) <sub>4~0</sub> =光标起始线 (CL) <sub>4~0</sub> =光标终止线	无
置光标位置	(AH)=2 (BH)=页号(图形模式为0) (DH)=行号 (DL)=列号	无
读当前光标位置	(AH)=3 (BH)=页号(图形模式为0)	(DH)=行号, (DL)=列号 (CX)=当前光标大小
读光笔位置	(AH)=4	(AH)=0, 未按光笔开关 (AH)=1, 按光笔开关 且下列寄存器值有效: (DH)=行号, (DL)=列号 图形方式时: (CH)=扫描线号(0~199) (BX)=像素列号(0~319 / 639)
置当前显示页(字符方式有效)	(AH)=5 (AL)=页号 模式0, 1=0~7 模式2, 3=0~3	无

功 能	入 口 参 数	出 口 参 数
上 滚 当 前 页	(AH)=6 (AL)=上滚行数, 0 为整个屏幕 (CH,CL)=滚动区域左上角的行、 列号 (DH,DL)=滚动区域右下角的行、 列号 (BH)=空白行的属性	无
下 滚 当 前 页	(AH)=7 (AL)=下滚行数(从窗口顶部算 起空白的行数) 0 为整个窗口空白 其他参数同上滚	无
读 当 前 光 标 位 置 处 的 字 符 及 属 性	(AH)=8 (BH)=页号(字符方式有效)	(AL)=读出的字符 (AH)=字符的属性
写 字 符 及 属 性 到 当 前 光 标 位 置 处	(AH)=9 (A1)=欲写字符 (BL)=字符属性 (CX)=字符计数 (BH)=页号(字符方式有效)	无
写 字 符 到 当 前 光 标 位 置 处 (属 性 不 变)	(AH)=10 (AL)=欲写字符 (CX)=字符计数 (BH)=页号(字符方式有效)	无
置 彩 色 调 色 板	(AH)=11 (BH)=调色板色别值 (BL)= 彩色值	无
在 屏 幕 上 写 点	(AH)=12 (DX)=线号, (CX)=列号 (AX)=点的颜色	无

功 能	入 口 参 数	出 口 参 数
在 屏 幕 上 读 点	(AH)=13 (DX)=线号, (CX)=列号	(AL)=点的颜色
写 字 符 到 当 前 光 标 位 置, 且 光 标 前 进 一 格	(AH)=14 (AL)=欲写字符 (BL)=字符颜色 (BH)=页号	无
读 当 前 显 示 状 态	(AH)=15	(AH)=屏幕上字符列 数 (AL)=当前显示模式 (BH)=当前显示页
保 留	((AH)=16 —18 )	
写字符串	(AH)=19 (ES: BP)=指向字符串 (CX)=字符串的长度 (DX)=起始的光标位置 (BH)=页号 (AL)=0, BL=属性 (字符, 字符, ……). 光标不移 动 (AL)=1, BL=属性 (字符, 字符, ……), 光标移动 (AL)=2, (字符, 属性, ……). 光 标不移动 (AL)=3, 同上. 光标移动	无

字符的属性字节含意如下:



例 红底蓝字的属性字节: 01000001 =41H  
 蓝底黄字的属性字节: 00010110 =16H