

# 实验十 键盘中断

学号：61822313

姓名：钟锦程

实验日期：2024 年 5 月 13 日

## 1 实验任务和实验结果

### 1.1 基础任务

#### 1.1.1 实验任务的具体内容

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。

#### 1.1.2 调试通过的源程序

```
1 DATA SEGMENT
2     OKSTR DB 'OK!',0DH,0AH,'$'
3     TIME DB 00H
4     NINE EQU 9
5 DATA ENDS;TIME是按键次数
6 STACK SEGMENT
7     DW 200H DUP(?)
8 STACK ENDS
9 CODE SEGMENT
10    ASSUME CS:CODE,DS:DATA,SS:STACK
11 DELAY PROC ; 延时程序
12     PUSH CX
13     PUSH DX
14     MOV DX,36H
15 DL500:
16     MOV CX, 08FFFH
17 DL10MS:
18     LOOP DL10MS
19     DEC DX
20     JNZ DL500
```

```
21     POP DX
22     POP CX
23     RET
24 DELAY ENDP
25
26 DELAY2 PROC ; 延时程序2,用于显示OK时的延时, 略短于1秒
27     PUSH CX
28     PUSH DX
29     MOV DX,16H
30 DL5002:
31     MOV CX, 01FFFH
32 DL10MS2:
33     LOOP DL10MS2
34     DEC DX
35     JNZ DL5002
36     POP DX
37     POP CX
38     RET
39 DELAY2 ENDP
40 DISP1 PROC FAR ; 显示太阳
41     PUSH AX
42     PUSH BX
43     PUSH CX
44     PUSH DX
45     MOV AH,15 ; 读当前显示状态
46     INT 10H
47     MOV AH,0 ; 设置显示方式
48     INT 10H
49     MOV DX,0 ; 行号为0, 列号为0
50 REPT:
51     MOV AH,2 ; 设置光标位置
52     INT 10H
53     MOV AL,0FH ; 0FH是太阳图形的ASCII码
54     MOV CX,1 ; 重复字符的次数
55     MOV AH,10 ; 写字符
56     INT 10H
57     CALL DELAY
58     SUB AL,AL
59     MOV AH,0 ; 清除原图形
```

```
60     INT 10H
61     INC DH ; 行号+1
62     ADD DL,2 ; 列号+2
63     CMP DH,20 ; 判断是否到20行, 不等继续显示太阳, 相等返回, 如果到25行再返回, 会
        导致轨迹不是严格对角
64     JNE REPT
65     POP DX
66     POP CX
67     POP BX
68     POP AX
69     RET
70 DISP1 ENDP
71 DISP2 PROC FAR ; 显示OK
72     PUSH CX
73     PUSH BX
74     PUSH AX
75     MOV CX,3;待显示的字符数
76 NEXTC:
77     LODSB ; 字符串"OK!"在数据段中定义, AL<-[SI]
78     MOV AH, 0EH ; 用teletype格式写字符, 并移动光标
79     MOV BX,1
80     INT 10H
81     CALL DELAY2
82     LOOP NEXTC
83     POP AX
84     POP BX
85     POP CX
86     RET
87 DISP2 ENDP
88
89 KEYINT PROC FAR
90     PUSH AX
91     PUSH BX
92     PUSH DX
93     STI
94     IN AL,60H ; 通过8255A的PA口(PA口地址为60H)读取键盘扫描码
95     MOV AH,AL
96     IN AL,61H ; 从8255APB口(PB口地址为61H)的PB7输出一个正脉冲
97     OR AL,80H ; PB7置1
```

```
98     OUT 61H,AL
99     AND AL,7FH ; PB7清零
100    OUT 61H,AL ;再输出一个负脉冲
101    TEST AH,80H ;相等时代表键被释放, 开中断, 显示字符
102    JNE BACK ; 不等, 中断结束返回
103    STI
104    INC TIME
105    MOV SI,OFFSET OKSTR ; 初始化SI, 在DISP2中不再对SI进行初始化
106    CALL DISP2
107 BACK:
108     MOV AL,20H
109     OUT 20H,AL
110     POP DX
111     POP BX
112     POP AX
113     IRET
114     KEYINT ENDP
115 START:
116     MOV AX,DATA
117     MOV DS,AX
118     MOV AX,STACK
119     MOV SS,AX
120     MOV AX,0
121     MOV ES,AX
122     MOV SI,OFFSET OKSTR
123     MOV AX,ES:[24H] ;  $9*4=24H$ ,压入中断向量的偏移地址
124     PUSH AX
125     MOV AX,ES:[26H] ;压入中断向量的段地址
126     PUSH AX
127     CLI;关中断, 载入自定义中断子程序地址时不允许中断
128     MOV AX,SEG KEYINT
129     MOV ES:[26H],AX
130     MOV AX,OFFSET KEYINT
131     MOV ES:[24H],AX
132     STI
133 AGIN:
134     CALL DISP1
135     CMP TIME,10
136     JB AGIN
```

```
137     CLI ; 禁止下方程序中断发生, 保护代码运行
138     POP AX
139     MOV ES:[26H],AX;弹出原中断向量的段地址
140     POP AX
141     MOV ES:[24H],AX;弹出原中断向量的偏移地址
142     STI ; 开中断
143     MOV AH,4CH
144     INT 21H
145 CODE ENDS
146 END START
```

### 1.1.3 实验结果

如图1是程序正常运行的截图, 如果不按键, 则当图标走到右下角后会从左上角重新开始, 循环往复。图2是按一次键的运行截图。可以看到在本应显示太阳符号的地方显示”OK!”, 如果多次按键(或长按), 会不间断显示”OK!”(如图3), 直到所有”OK!”显示完毕才会继续显示太阳符号。当按键总次数达到 10 次时, 程序会在本次太阳显示循环结束后中止程序, 如图4。

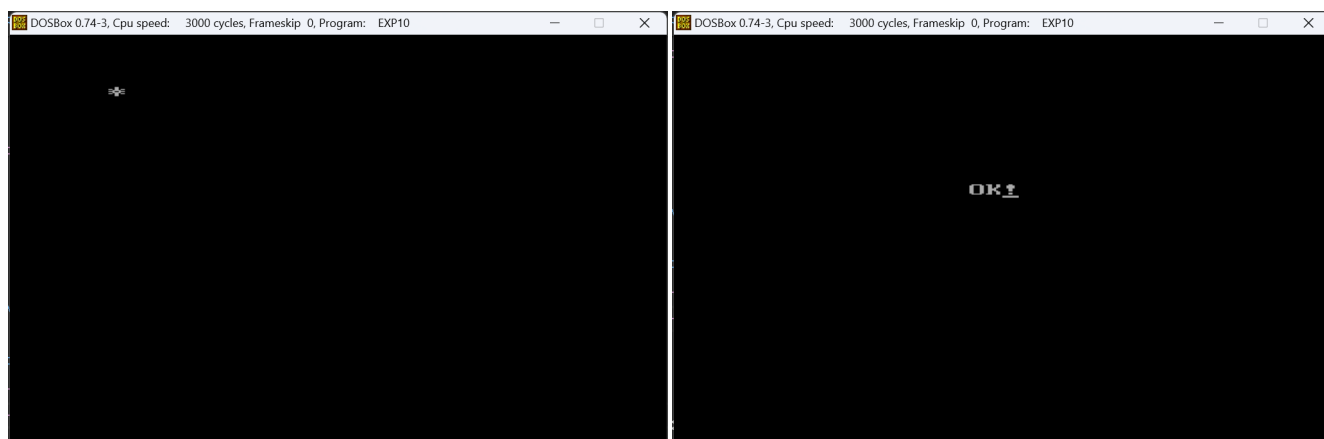


图 1: 基础实验任务结果截图 1

图 2: 基础实验任务结果截图 2

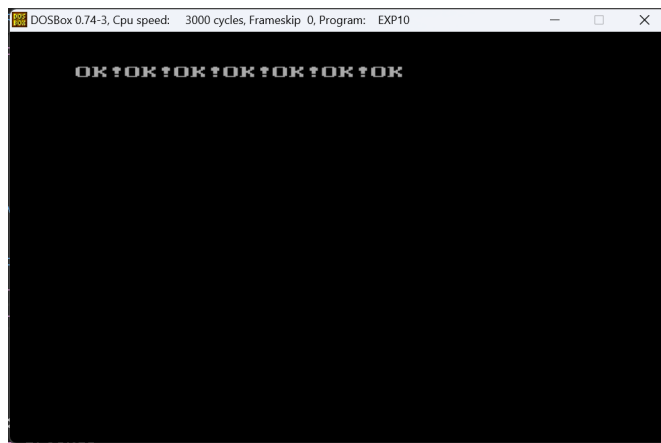


图 3: 基础实验任务结果截图 1

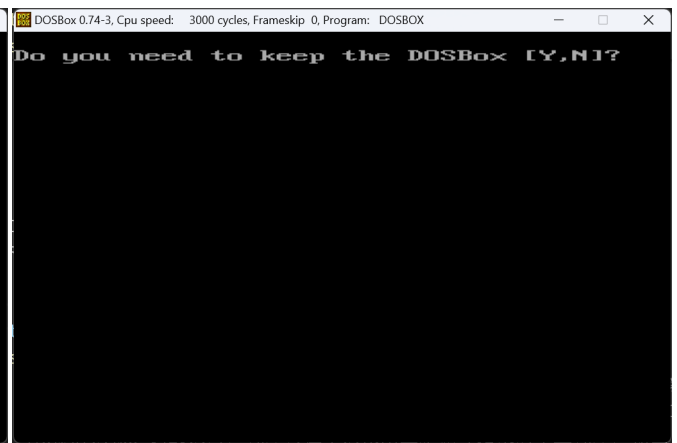


图 4: 基础实验任务结果截图 2

## 1.2 附加任务

### 1.2.1 实验任务的具体内容

- 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

### 1.2.2 调试通过的源程序

```

1 DATA SEGMENT
2     TIME DB 00H
3     SHOWTIME DB 00H;SHOWTIME存放TIME对应的ASCII码
4     OKSTR DB 'HI,ZJC!',20H
5     N EQU $-SHOWTIME
6     COUNT DB N;COUNT用于在显示过程中计数递减, 便于直接中断而不用等本次太阳循
        环结束
7     NINE EQU 9;定义数值常量便于调试
8 DATA ENDS;TIME是按键次数
9 STACK SEGMENT
10     DW 200H DUP(?)
11 STACK ENDS

```

```
12 CODE SEGMENT
13     ASSUME CS:CODE,DS:DATA,SS:STACK
14 DELAY PROC ; 延时程序
15     PUSH CX
16     PUSH DX
17     MOV DX,36H
18 DL500:
19     MOV CX, 08FFFH
20 DL10MS:
21     LOOP DL10MS
22     DEC DX
23     JNZ DL500
24     POP DX
25     POP CX
26     RET
27 DELAY ENDP
28
29 DELAY2 PROC ; 延时程序2
30     PUSH CX
31     PUSH DX
32     MOV DX,16H
33 DL5002:
34     MOV CX, 01FFFH
35 DL10MS2:
36     LOOP DL10MS2
37     DEC DX
38     JNZ DL5002
39     POP DX
40     POP CX
41     RET
42 DELAY2 ENDP
43
44 DISP1 PROC FAR ; 显示太阳
45     PUSH AX
46     PUSH BX
47     PUSH CX
48     PUSH DX
49     MOV AH,15 ; 读当前显示状态
50     INT 10H
```

```
51     MOV AL,1
52     MOV AH,0 ; 设置显示方式
53     INT 10H
54     MOV DX,0 ; 行号为0, 列号为0
55 REPT:
56     MOV AH,2 ; 设置光标位置
57     INT 10H
58     MOV AL,0FH ; 0FH是太阳图形的ASCII码
59     MOV CX,1 ; 重复字符的次数
60     MOV BL,14H ; 字符颜色信息为白底红字加闪烁 (b7控制字符是否闪烁, b6-b4为背景
        色, b3-b0为前景色)
61     MOV AH,9 ; 写字符
62     INT 10H
63     CALL DELAY
64     SUB AL,AL
65     MOV AH,0 ; 清除原图形
66     INT 10H
67     CMP TIME,NINE;当次数达到设定值时, 直接结束显示
68     JAE QUIT1
69     INC DH ; 行号+1
70     ADD DL,2 ; 列号+2
71     CMP DH,20 ; 判断是否到20行, 不等则继续显示太阳, 相等返回
72     JNE REPT
73 QUIT1:
74     POP DX
75     POP CX
76     POP BX
77     POP AX
78     RET
79 DISP1 ENDP
80
81 DISP2 PROC FAR ; 中断显示信息
82 ; INT 10H 中AH=03H读取光标信息。入口参数: BH=显示页码
83 ; 出口参数: CH=光标的起始行 CL=光标的终止行 DH=行(Y坐标) DL=列 (X坐标)
84     PUSH DX
85     PUSH CX
86     PUSH BX
87     PUSH AX
88     MOV COUNT,N;用待显示的字符串的长度更新COUNT
```



```
89 NEXTC:
90     LODSB ; 字符串在数据段中定义并已经由SI指向首址
91     MOV BH,00H
92     MOV AH,03H
93     INT 10H
94     MOV AH,9 ;写字符
95     MOV CX,1 ;重复一次
96     MOV BL,24H ;字符颜色信息为绿底红字
97     INT 10H
98     INC DL ;每打印1个字符，光标的列号+1
99     MOV AH,2 ;更新光标位置
100    INT 10H
101    CALL DELAY2
102    DEC COUNT
103    CMP COUNT,0 ;当整个中断信息串都显示完，才退出循环
104    JA NEXTC
105    POP AX
106    POP BX
107    POP CX
108    POP DX
109    RET
110 DISP2 ENDP
111
112 KEYINT PROC FAR
113     PUSH AX
114     PUSH BX
115     PUSH DX
116     STI
117     IN AL,60H ;通过8255A的PA口(PA口地址为60H)读取键盘扫描码
118     MOV AH,AL
119     IN AL,61H ;从8255APB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)
120     OR AL,80H ;PB7置1
121     OUT 61H,AL
122     AND AL,7FH; PB7清零
123     OUT 61H,AL
124     TEST AH,80H ;相等时代表键被释放，开中断，显示字符
125     JNE BACK
126     STI
```

```
127     INC TIME
128     MOV AL,0
129     ADD AL,TIME
130     OR AL,30H
131     MOV SI,OFFSET SHOWTIME ; 初始化SI,使其指向中断信息字符串首址
132     MOV [SI],AL
133     CALL DISP2
134 BACK:
135     MOV AL,20H
136     OUT 20H,AL;结束中断
137     POP DX
138     POP BX
139     POP AX
140     IRET
141     KEYINT ENDP
142
143 START:
144     MOV AX,DATA
145     MOV DS,AX
146     MOV AX,STACK
147     MOV SS,AX
148     MOV AX,0
149     MOV ES,AX
150     MOV SI,OFFSET TIME
151     MOV AH,35H ;INT 21H 35H号: 取中断向量, 入口AL=中断类型, 出口ES:BX=中
        断向量
152     MOV AL,9
153     INT 21H
154     PUSH BX ;将原中断处理程序的CS:IP压栈
155     PUSH ES
156     MOV AX,0 ;恢复ES!重要!
157     MOV ES,AX
158     PUSH DS ;保护DS与DX!重要!
159     PUSH DX
160     CLI
161     MOV AX,SEG KEYINT ;将自定义中断服务程序的地址放入原09号中断向量处
162     MOV DS,AX
163     MOV DX,OFFSET KEYINT
164     MOV AL,9
```

```
165     MOV AH,25H ;INT 21H 25H号：设置中断向量，入口DS:DX=中断向量，AL=中断
        类型号
166     INT 21H
167     STI
168     POP DX
169     POP DS
170 AGIN:
171     CALL DISP1
172     CMP TIME,NINE
173     JB AGIN
174     CLI ;禁止下方程序中断发生，保护代码运行
175     POP DS ;恢复原09号中断向量
176     POP DX
177     MOV AL,9
178     MOV AH,25H
179     INT 21H
180     STI ;开中断
181     MOV AH,4CH
182     INT 21H
183 CODE ENDS
184 END START
```

### 1.2.3 实验结果

如图5是程序正常运行的截图，太阳符号被修改为白底红字加闪烁（附加功能 5、6）。图6是按一次键的运行截图。显示了绿底红字的中断信息“HI!ZJC”并在该字符串前面显示了按键次数（附加功能 2）。图7是长时间按键的截图，程序不断打印中断信息，信息之间有空格（附加功能 3）。图8是按键次数达到 9 次时的截图，可以看到程序直接退出返回 DOS，而非等到本次太阳显示完才结束（附加功能 4）。（附加功能 1 可从程序源代码中看到）

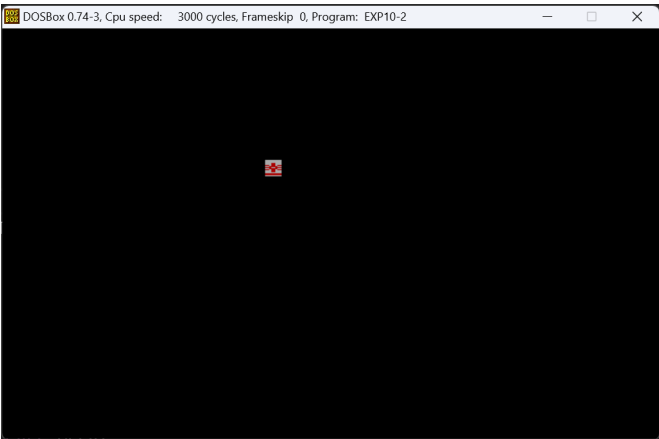


图 5: 附加实验任务结果截图 1

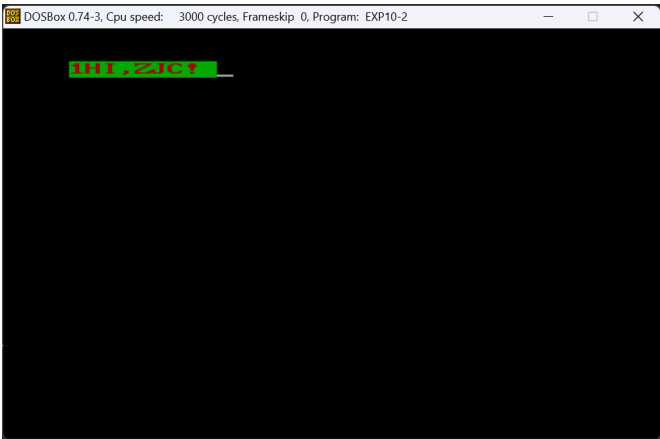


图 6: 附加实验任务结果截图 2



图 7: 附加实验任务结果截图 3



图 8: 附加实验任务结果截图 4

## 2 实验总结

本次实验首次接触键盘中断，对自定义中断服务程序的流程不熟悉，并且还没有形成保护现场的习惯。在用 DOS 中断服务 (AH=25H INT 21H) 编写中断服务程序 KEYINT 时，由于没有保护 ES 的值，导致读取中断向量后 ES 被改变，中断程序跳转错误。在输入键扫描码，键状态触发器复位的设计过程中也遇到了不小的困难。

另外，在本次实验中较为细致的学习了屏幕显示的相关知识，比如设置显示参数，控制光标等等。编写程序时比较麻烦的一点就是对光标的控制。如果要自定义字符属性就不能使用 Teletype 方法打印，而必须自己手动控制光标位置。

## 3 思考题

1. 当键盘上的一个按键按下时，键盘会发送一个中断信号给 CPU，与此同时，键盘会在指定端口 (0x60) 输出一个数值 a，当按键弹起时，键盘又给端口输出一个数值 b。通过查

表得, a 的最高位都是 0, b 的最高位都是 1, 因此若按键释放, 指令 TEST AH,80H 使 ZF=1, 后续 JNE 指令跳转到 BACK, 直接退出 KEYINT, 不显示字符。

2. 附加功能如何实现: 修改 DISP1 子程序, 在 REPT 部分加入指令 CMP TIME,NINE 和 JAE QUIT1, 当按键达到指定次数时直接跳转到退出部分, 而不需等待本次显示结束。
3. 键盘不能正常使用的可能原因: 在程序末尾没有恢复原 09 号中断向量的内容, 导致键盘中断处理异常。