

计算机组成原理 Homework17 (12.28)

中国人民大学 信息学院 崔冠宇 2018202147

1. 参考课堂上的硬件时钟中断的例子 hintr.com，编写程序实现以下功能：

程序运行之后，判断秒显示标志，标志有效时，显示 MM:SS。其中 MM 表示分钟，SS 表示秒。要求编写中断处理程序，按照时钟的计数方式计数。主程序在按任意键时，退出。

解：在 200H 起的单元按下列格式设置数据：

- 200H-203H: 存放旧中断向量
- 204H: 秒计数器，到 18 为 1 秒
- 205H: 显示标记，00 不显示，FF 显示
- 206H 起: “00:00”字符串（以及后面的换行和 \$），用来输出

-d 200																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																				
--------	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

图 1: 数据区

```
1 MAIN: ; 100H 起输入主程序
2 MOV AX, 351C
3 INT 21 ; 35H 系统调用获取原中断向量
4 CS:
5 MOV [200], DX
6 CS:
7 MOV [202], ES ; 保护原中断向量
8 MOV DX, 220
9 MOV AX, 251C
10 INT 21 ; 25H 系统调用设置新中断向量
11 LP0: (0117)
12 MOV AL, [205]
13 CMP AL, FF ; 判断标志是否有效(是否到了 1 秒)
14 JNZ LP1(012A)
15 MOV AL, 0
16 MOV [205], AL ; 标志有效，重设标志
17 MOV DX, 206
18 MOV AH, 9
```

```
19 INT 21 ; 9H 显示内容
20 LP1: (012A)
21 MOV AH, B
22 INT 21 ; BH 系统调用获取键盘输入
23 CMP AL, 0
24 JZ LP0(0117) ; 无键盘输入则继续
25 CS:
26 LDS DX, [200]
27 MOV AX, 251C
28 INT 21 ; 25H 系统调用恢复中断向量
29 INT 20
30
31 INTERRUPT: ; 220H 起输入中断服务程序
32 PUSH AX
33 MOV AL, [204]
34 INC AL
35 CMP AL, 12 ; 判断是否到 12H, 即 1 秒
36 JNZ LP2(281) ; 228
37 MOV AL, FF
38 MOV [205], AL ; 设置标志有效
39 MOV AL, [20A] ; 获取秒最低位
40 INC AL ; 增加 1 秒
41 CMP AL, 3A ; 超过 10 了
42 JNZ LP3(244) ; 236
43 SUB AL, A ; 减去 10
44 PUSH AX ; 保护 AX
45 MOV AL, [209]
46 INC AL ; 次低位加 1
47 MOV [209], AL ; 写回
48 POP AX ; 弹出 AX
49 LP3: (244)
50 MOV [20A], AL ; 写回最低位
51 MOV AL, [209]
52 CMP AL, 36 ; 判断秒次低位是否超过 6
53 JNZ LP4(25D) ; 24C
54 SUB AL, 6 ; 减去 6
55 PUSH AX
```

```

56 MOV AL, [207]
57 INC AL                ; 分钟数加 1
58 MOV [207], AL        ; 写回
59 POP AX
60 MOV [209], AL        ; 写回秒次低位
61 LP4: (25D)
62 MOV AL, [207]
63 CMP AL, 3A           ; 判断分钟数最低位是否超过 10
64 JNZ LP5(273)         ; 262
65 SUB AL, A            ; 减去 10
66 PUSH AX              ; 保护 AX
67 MOV AL, [206]
68 INC AL               ; 分钟次低位加 1
69 MOV [206], AL        ; 写回
70 POP AX               ; 弹出 AX
71 MOV [207], AL        ; 写回分钟最低位
72 LP5: (273)
73 MOV AL, [206]
74 CMP AL, 36           ; 判断分钟最高位是否达到 6
75 JNZ LP6(27F)
76 SUB AL, 6
77 MOV [206], AL        ; 减 6 写回
78 LP6: (27F)
79 MOV AL, 0
80 LP2: (281)
81 MOV [204], AL        ; 写回 AL
82 POP AX
83 IRET

```

在 DEBUG 环境中，输入上述代码，设置数据区并运行。部分截图如下所示：

```
DOSBox 0.74-3-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
075A:0251 A00702 MOV AL,[0207]
075A:0254 FEC0 INC AL
075A:0256 A20702 MOV [0207],AL
075A:0259 58 POP AX
075A:025A A20902 MOV [0209],AL
075A:025D A00702 MOV AL,[0207]
075A:0260 3C3A CMP AL,3A
075A:0262 750F JNZ 0273
075A:0264 2C0A SUB AL,0A
075A:0266 50 PUSH AX
075A:0267 A00602 MOV AL,[0206]
075A:026A FEC0 INC AL
075A:026C A20602 MOV [0206],AL
075A:026F 58 POP AX
075A:0270 A20702 MOV [0207],AL
075A:0273 A00602 MOV AL,[0206]
075A:0276 3C36 CMP AL,36
075A:0278 7505 JNZ 027F
075A:027A 2C06 SUB AL,06
075A:027C A20602 MOV [0206],AL
075A:027F B000 MOV AL,00
075A:0281 A20402 MOV [0204],AL
075A:0284 58 POP AX
075A:0285 CF IRET
```

图 2: 键入代码

```
DOSBox 0.74-3-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
00:46
00:47
00:48
00:49
00:50
00:51
00:52
00:53
00:54
00:55
00:56
00:57
00:58
00:59
01:00
01:01
01:02
01:03
01:04
01:05
01:06
Program terminated normally
-ú
```

图 3: 计时器工作正常，按任意键可以退出

根据程序运行结果可见代码功能正确。

2. 参考课堂上的软中断的例子 sintr.com，编写程序实现以下功能：程序运行之后，按“A”键，触发中断程序，计数器内容加 1，按“B”键，触发中断程序，计数器内容减 1。无论加 1 和减 1 均限制在 ascii 码：30~39 之间。按“Q”键时，退出。

解：在 200H 起的单元按下列格式设置数据：

- 200H-203H: 存放旧中断向量
- 204H: 键盘输入的控制字符
- 205H: 计数器，计数 30~39（以及后面的换行和 \$）

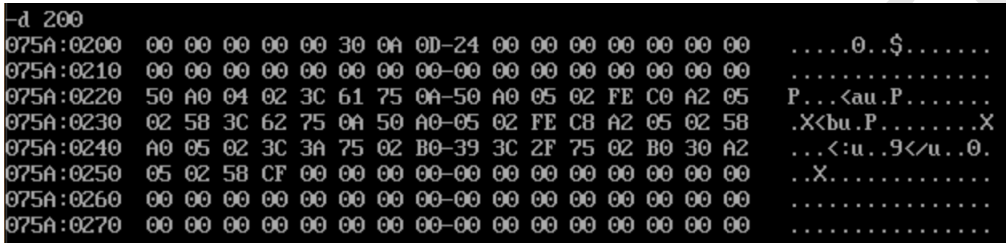


图 4: 数据区

```
1 MAIN: ; 100H 起输入主程序
2 MOV AX, 356B
3 INT 21 ; 35H 系统调用获取原中断向量
4 CS:
5 MOV [200], BX
6 CS:
7 MOV [202], ES ; 保护原中断向量
8 MOV DX, 220
9 MOV AX, 256B
10 INT 21 ; 25H 系统调用设置新中断向量
11 LP0: (117)
12 MOV AH, 1
13 INT 21 ; 读一个字符
14 CMP AL, 71 ; 若是 'q', 退出
15 JZ LP1(12D) ; 11D
16 MOV [204], AL ; 键盘字符放到 204H
17 INT 6B ; 软中断 —— 根据键盘内容修改计数器
18 MOV DX, 205
19 MOV AH, 9
20 INT 21 ; 显示计数器内容
21 JMP LP0(117)
22 LP1: (12D)
```

```

23 CS:
24 LDS DX, [200]
25 MOV AX, 256B
26 INT 21 ; 25H 系统调用恢复中断向量
27 INT 20
28
29 INTERRUPT: ; 220H 起输入中断服务程序
30 PUSH AX
31 MOV AL, [204] ; 获取输入的字符
32 CMP AL, 61 ; 'a' = 61H
33 JNZ LP2(232) ; 226
34 PUSH AX ; 保护 AX
35 MOV AL, [205]
36 INC AL
37 MOV [205], AL ; 加 1, 写回
38 POP AX ; 弹出 AX
39 LP2: (232)
40 CMP AL, 62 ; 'b' = 62H
41 JNZ LP3(240) ; 234
42 PUSH AX ; 保护 AX
43 MOV AL, [205]
44 DEC AL
45 MOV [205], AL ; 减少 1, 写回
46 POP AX ; 弹出 AX
47 LP3: (240)
48 MOV AL, [205]
49 CMP AL, 3A ; 大了
50 JNZ LP4(249) ; 245
51 MOV AL, 39 ; 重设为 39H
52 LP4: (249)
53 CMP AL, 2F ; 小了
54 JNZ LP5(24F) ; 24B
55 MOV AL, 30 ; 重设为 30H
56 LP5: (24F)
57 MOV [205], AL ; 写回
58 POP AX
59 IRET

```

在 DEBUG 环境中，输入上述代码，设置数据区并运行。部分截图如下所示：

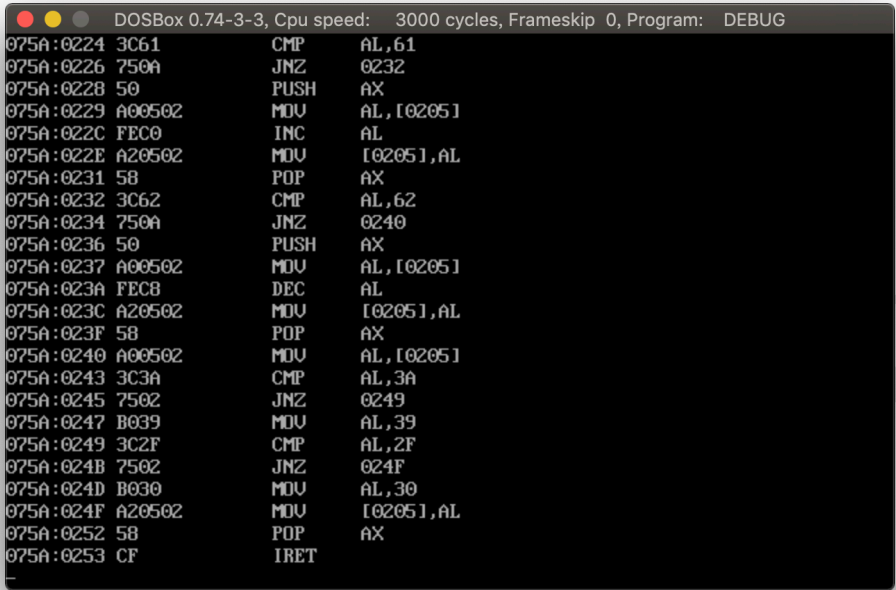


图 5: 键入代码

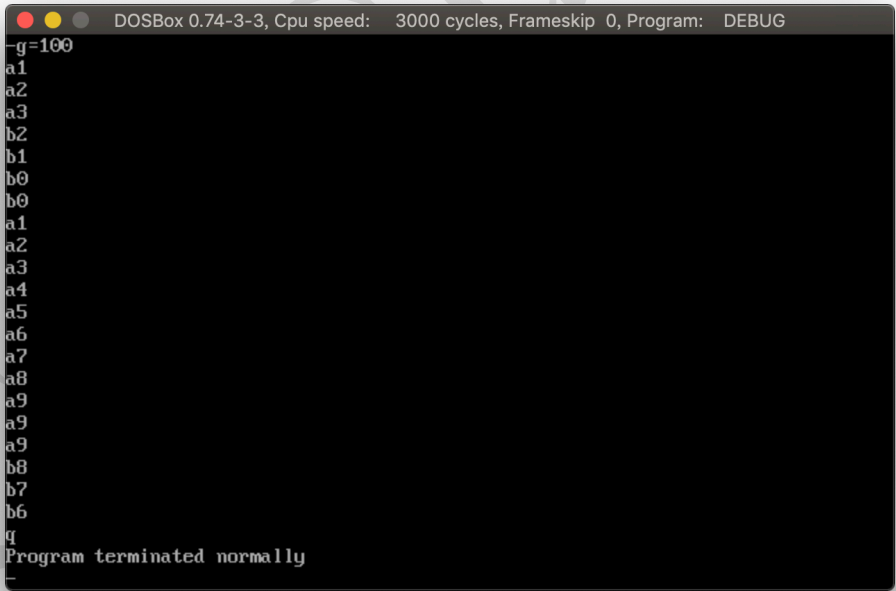


图 6: 按下 a 计数器 +1，按下 b 计数器-1（不会超出范围），按下 q 退出

根据程序运行结果可见代码功能正确。