

汇编语言第4次上机

班级	学号	姓名
计算机2205	2204112913	李雨轩

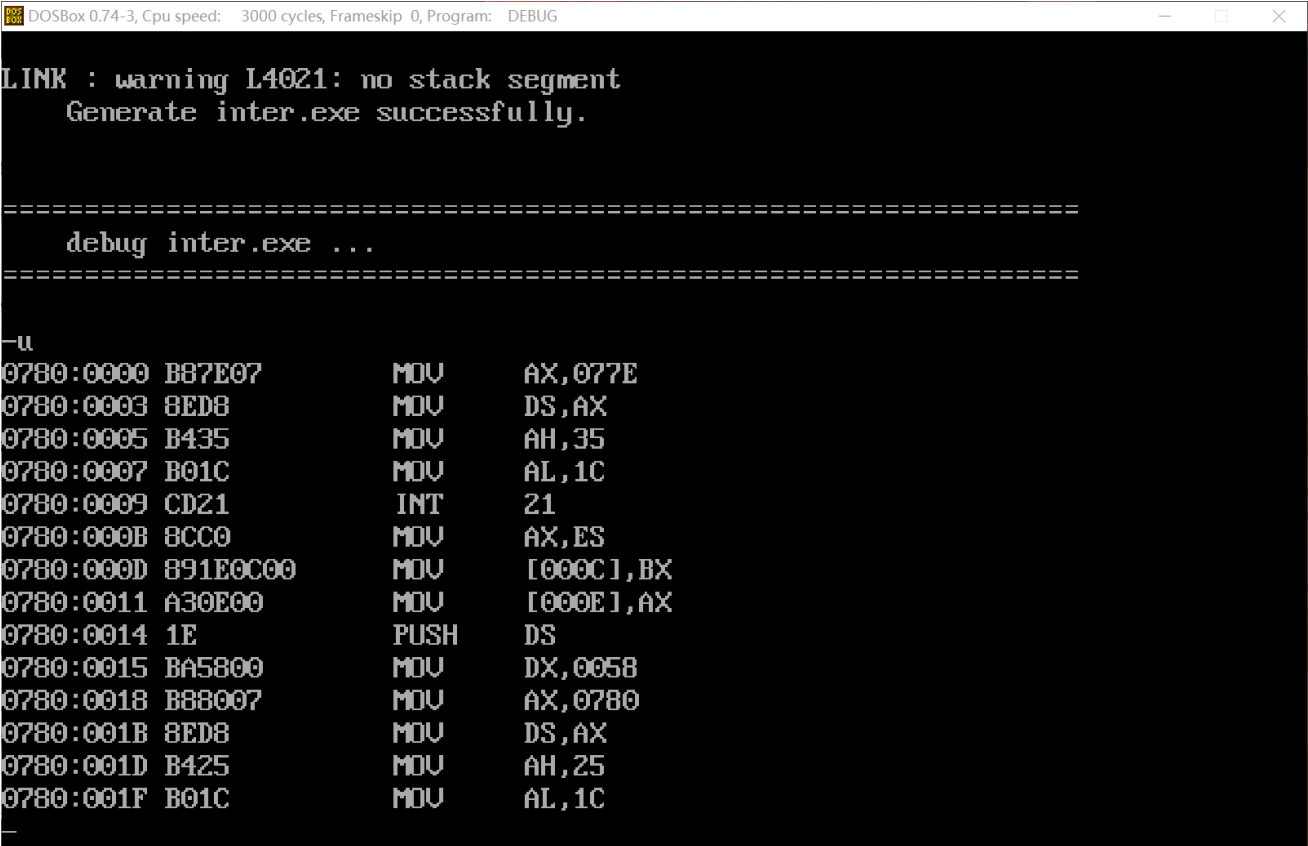
1. 中断子程序设计

编写程序，使类型 1CH 的中断向量指向中断处理程序 COUNT，COUNT 统计 1CH 中断次数并存入字变量单元 NUM 中。程序启动后等待用户输入，输入字符 Q 后退出，并将 NUM 值用十六进制形式显示出来。例如 NUM 的内容为 1234h，则在屏幕上显示 1234h。

数据段中至少需要定义以下内容：

- (1) ID db '2186123456' (说明：以学号2186123456为例)
- (2) 定义中断次数 NUM 的内存单元

(1) 反汇编的截图



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
0780:0011 A30E00      MOV     [000E],AX
0780:0014 1E          PUSH    DS
0780:0015 BA5800      MOV     DX,0058
0780:0018 B88007      MOV     AX,0780
0780:001B 8ED8          MOV     DS,AX
0780:001D B425          MOV     AH,25
0780:001F B01C          MOV     AL,1C
-u
0780:0021 CD21      INT     21
0780:0023 1F          POP     DS
0780:0024 E421      IN      AL,21
0780:0026 24FE      AND     AL,FE
0780:0028 E621      OUT     21,AL
0780:002A FB          STI
0780:002B B401      MOV     AH,01
0780:002D CD21      INT     21
0780:002F 3C51      CMP     AL,51
0780:0031 75FA      JNZ     002D
0780:0033 B409      MOV     AH,09
0780:0035 8D161000  LEA     DX,[0010]
0780:0039 CD21      INT     21
0780:003B 1E          PUSH    DS
0780:003C A10E00      MOV     AX,[000E]
0780:003F 8B160C00  MOV     DX,[000C]
-
```

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
0780:002D CD21      INT     21
0780:002F 3C51      CMP     AL,51
0780:0031 75FA      JNZ     002D
0780:0033 B409      MOV     AH,09
0780:0035 8D161000  LEA     DX,[0010]
0780:0039 CD21      INT     21
0780:003B 1E          PUSH    DS
0780:003C A10E00      MOV     AX,[000E]
0780:003F 8B160C00  MOV     DX,[000C]
-u
0780:0043 8ED8          MOV     DS,AX
0780:0045 B425          MOV     AH,25
0780:0047 B01C          MOV     AL,1C
0780:0049 CD21      INT     21
0780:004B 1F          POP     DS
0780:004C 8B1E0A00  MOV     BX,[000A]
0780:0050 E81300      CALL    0066
0780:0053 B8004C      MOV     AX,4C00
0780:0056 CD21      INT     21
0780:0058 50          PUSH    AX
0780:0059 B87E07      MOV     AX,077E
0780:005C 8ED8          MOV     DS,AX
0780:005E FB          STI
0780:005F FF060A00  INC     WORD PTR [000A]
-
```

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
0780:0056 CD21      INT     21
0780:0058 50        PUSH    AX
0780:0059 B87E07      MOV     AX,077E
0780:005C 8ED8      MOV     DS,AX
0780:005E FB        STI
0780:005F FF060A00    INC     WORD PTR [000A]
-u
0780:0063 FA        CLI
0780:0064 58        POP     AX
0780:0065 CF        IRET
0780:0066 50        PUSH    AX
0780:0067 51        PUSH    CX
0780:0068 52        PUSH    DX
0780:0069 B90400      MOV     CX,0004
0780:006C D1C3      ROL     BX,1
0780:006E D1C3      ROL     BX,1
0780:0070 D1C3      ROL     BX,1
0780:0072 D1C3      ROL     BX,1
0780:0074 8AD3      MOV     DL,BL
0780:0076 80E20F      AND     DL,0F
0780:0079 80FA09      CMP     DL,09
0780:007C 7F06      JG      0084
0780:007E 80C230      ADD     DL,30
0780:0081 EB04      JMP     0087
-S_
```

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
0780:0079 80FA09      CMP     DL,09
0780:007C 7F06      JG      0084
0780:007E 80C230      ADD     DL,30
0780:0081 EB04      JMP     0087
-u
0780:0083 90        NOP
0780:0084 80C237      ADD     DL,37
0780:0087 B402      MOV     AH,02
0780:0089 CD21      INT     21
0780:008B E2DF      LOOP    006C
0780:008D B268      MOV     DL,68
0780:008F B402      MOV     AH,02
0780:0091 CD21      INT     21
0780:0093 5A        POP     DX
0780:0094 59        POP     CX
0780:0095 58        POP     AX
0780:0096 C3        RET
0780:0097 4E        DEC     SI
0780:0098 42        INC     DX
0780:0099 3030      XOR     [BX+SI],DH
0780:009B 350200      XOR     AX,0002
0780:009E 0000      ADD     [BX+SI],AL
0780:00A0 0000      ADD     [BX+SI],AL
0780:00A2 0000      ADD     [BX+SI],AL
_
```

(2) 在进行计算前，显示 ID、NUM 的内存值的截图（多显示、少显示均扣分）

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
0780:0024 E421      IN      AL,21
-u
0780:0026 24FE      AND     AL,FE
0780:0028 E621      OUT     21,AL
0780:002A FB         STI
0780:002B B401      MOV     AH,01
0780:002D CD21      INT     21
0780:002F 3C51      CMP     AL,51
0780:0031 75FA      JNZ     002D
0780:0033 B409      MOV     AH,09
0780:0035 8D161000    LEA     DX,[0010]
0780:0039 CD21      INT     21
0780:003B 1E         PUSH    DS
0780:003C A10E00      MOV     AX,[000E]
0780:003F 8B160C00    MOV     DX,[000C]
0780:0043 8ED8      MOV     DS,AX
0780:0045 B425      MOV     AH,25
-g5

AX=077E BX=0000 CX=031E DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=077E ES=076E SS=077D CS=0780 IP=0005  NU UP EI PL NZ NA PO NC
0780:0005 B435      MOV     AH,35
-d 077e:0 B
077E:0000 32 32 30 34 31 31 32 39-31 33 00 00          2204112913..
-
```

(3) 运行到返回 dos 前暂停，对屏幕显示的输出结果（NUM 值的对应的 ASCII 字符串）截图【结果要与步骤（4）中的内存值一致】

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
0780:002D CD21      INT     21
0780:002F 3C51      CMP     AL,51
0780:0031 75FA      JNZ     002D
0780:0033 B409      MOV     AH,09
0780:0035 8D161000    LEA     DX,[0010]
0780:0039 CD21      INT     21
0780:003B 1E         PUSH    DS
0780:003C A10E00      MOV     AX,[000E]
0780:003F 8B160C00    MOV     DX,[000C]
0780:0043 8ED8      MOV     DS,AX
0780:0045 B425      MOV     AH,25
-g5

AX=077E BX=0000 CX=031E DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=077E ES=076E SS=077D CS=0780 IP=0005  NU UP EI PL NZ NA PO NC
0780:0005 B435      MOV     AH,35
-d 077e:0 B
077E:0000 32 32 30 34 31 31 32 39-31 33 00 00          2204112913..
-g56
2204112913 li yuxuan in XJTU Q
01C2h
AX=4C00 BX=01C2 CX=031E DX=1260 SP=0000 BP=0000 SI=0000 DI=0000
DS=077E ES=F000 SS=077D CS=0780 IP=0056  NU UP EI PL NZ NA PO NC
0780:0056 CD21      INT     21
-
```

(4) 在完成步骤 (3) 操作后，立即显示 ID、NUM 的内存值的截图（多显示、少显示均扣分）

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
0780:0031 75FA      JNZ      002D
0780:0033 B409      MOV      AH,09
0780:0035 8D161000    LEA      DX,[0010]
0780:0039 CD21      INT      21
0780:003B 1E        PUSH     DS
0780:003C A10E00      MOV      AX,[000E]
0780:003F 8B160C00    MOV      DX,[000C]
0780:0043 8ED8      MOV      DS,AX
0780:0045 B425      MOV      AH,25
-g5
AX=077E BX=0000 CX=031E DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=077E ES=076E SS=077D CS=0780 IP=0005  NU UP EI PL NZ NA PO NC
0780:0005 B435      MOV      AH,35
-d 077e:0 B
077E:0000 32 32 30 34 31 31 32 39-31 33 00 00                2204112913..
-g56
2204112913 li yuxuan in XJTU Q
01C2h
AX=4C00 BX=01C2 CX=031E DX=1260 SP=0000 BP=0000 SI=0000 DI=0000
DS=077E ES=F000 SS=077D CS=0780 IP=0056  NU UP EI PL NZ NA PO NC
0780:0056 CD21      INT      21
-d 077e:0 B
077E:0000 32 32 30 34 31 31 32 39-31 33 C2 01                2204112913..
_
```

(5) 源代码

```
1  name Interrupt
2  title Int sub-program
3
4  assume cs:code, ds:data
5
6  addr struc
7      offsets dw ?
8      segments dw ?
9  addr ends
10
11 data segment
12     ID db '2204112913'
13     NUM dw 0
14     IADDR addr <>
15     crlf db 0DH, 0AH, '$'
16 data ends
17
18 code segment
19     main proc far
20         mov ax, seg data
21         mov ds, ax
22
```

```

23      ; get the interrupt vector
24      mov ah, 35h
25      mov al, 1ch
26      int 21h
27      mov ax, es
28      mov IADDR.offsets, bx
29      mov IADDR.segments, ax
30
31      ; set the interrupt vector
32      push ds
33      mov dx, offset COUNT
34      mov ax, seg COUNT
35      mov ds, ax
36      mov ah, 25h
37      mov al, 1ch
38      int 21h
39      pop ds
40
41      ; enable the timer tick interrupt
42      in al, 21h
43      and al, 11111110b
44      out 21h, al
45      sti
46
47      ; wait for the user to press 'Q'
48      mov ah, 1
49  waitForKey:
50      int 21h
51      cmp al, 'Q'
52      jne waitForKey
53
54      ;print crlf
55      mov ah, 09h
56      lea dx, crlf
57      int 21h
58
59      ; reset the interrupt vector
60      push ds
61      mov ax, IADDR.segments
62      mov dx, IADDR.offsets
63      mov ds, ax
64      mov ah, 25h
65      mov al, 1ch
66      int 21h
67      pop ds
68
69      mov bx, NUM
70      call HexShot
71
72      mov ax, 4c00H
73      int 21h

```

```

74
75     main endp
76
77     COUNT proc far
78         push ax
79         mov ax, data
80         mov ds, ax
81
82         sti
83         inc NUM
84         cli
85
86         pop ax
87         iret
88     COUNT endp
89
90     ; print the number in bx in hexadecimal
91     HexShot proc
92         push ax
93         push cx
94         push dx
95
96         mov cx, 4
97     print_digit:
98         rol bx, 1
99         rol bx, 1
100        rol bx, 1
101        rol bx, 1
102        mov dl, bl
103        and dl, 0Fh
104        cmp dl, 9
105        jg convert_letter
106        add dl, '0'
107        jmp print_done
108    convert_letter:
109        add dl, 'A' - 10
110    print_done:
111        mov ah, 02h
112        int 21h
113        loop print_digit
114
115        ;print "h"
116        mov dl, 'h'
117        mov ah, 02h
118        int 21h
119
120        pop dx
121        pop cx
122        pop ax
123        ret
124    HexShot endp

```

```
125 | code ends
126 |     end main
```

2. BIOS 和 DOS 中断

编写一个程序，接收从键盘输入的 10 个十进制数字（你的学号），输入回车符则停止输入，然后将这些数字加密后（用 `XLAT` 指令变换）存入内存缓冲区 `BUFFER`。加密表为：

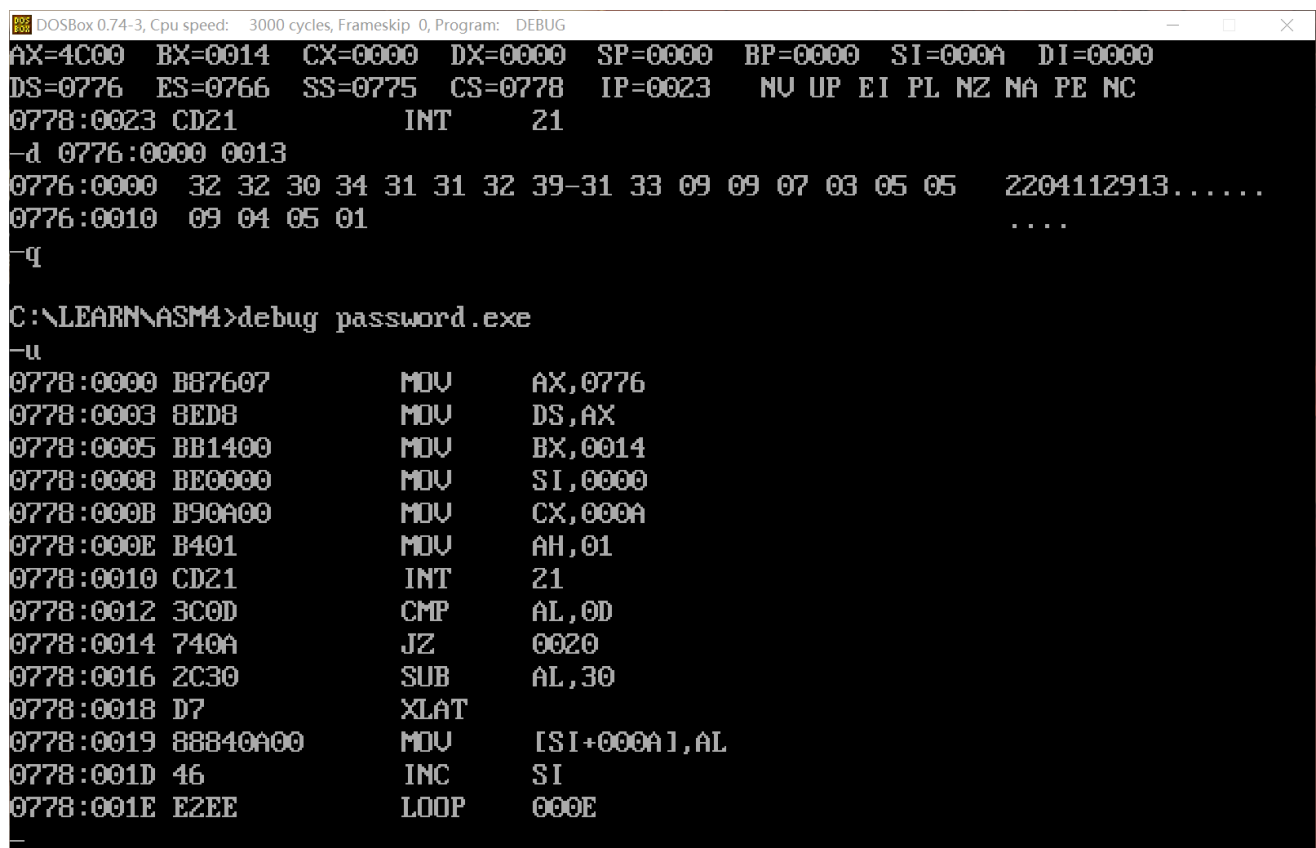
输入数字：0, 1, 2, 3, 4, 5, 6, 7, 8, 9

密码数字：7, 5, 9, 1, 3, 6, 8, 0, 2, 4

数据段中至少需要定义以下内容：

- (1) `ID db '2186123456'` (说明：以学号 2186123456 为例)
- (2) `BUFFER db 10 dup (?)`

(1) 反汇编的截图



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
AX=4C00 BX=0014 CX=0000 DX=0000 SP=0000 BP=0000 SI=000A DI=0000
DS=0776 ES=0766 SS=0775 CS=0778 IP=0023  NU UP EI PL NZ NA PE NC
0778:0023 CD21          INT     21
-d 0776:0000 0013
0776:0000 32 32 30 34 31 31 32 39-31 33 09 09 07 03 05 05  2204112913.....
0776:0010 09 04 05 01                                     ....
-q
C:\LEARN\ASM4>debug password.exe
-u
0778:0000 B87607          MOV     AX,0776
0778:0003 8ED8          MOV     DS,AX
0778:0005 BB1400          MOV     BX,0014
0778:0008 BE0000          MOV     SI,0000
0778:000B B90A00          MOV     CX,000A
0778:000E B401          MOV     AH,01
0778:0010 CD21          INT     21
0778:0012 3C0D          CMP     AL,0D
0778:0014 740A          JZ      0020
0778:0016 2C30          SUB     AL,30
0778:0018 D7          XLAT
0778:0019 88840A00          MOV     [SI+000A],AL
0778:001D 46          INC     SI
0778:001E EZEE          LOOP   000E
```



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
0778:0018 D7 XLAT
0778:0019 88840A00 MOV [SI+000A],AL
0778:001D 46 INC SI
0778:001E E2EE LOOP 000E
-u
0778:0020 B8004C MOV AX,4C00
0778:0023 CD21 INT 21
0778:0025 4E DEC SI
0778:0026 42 INC DX
0778:0027 3030 XOR [BX+SI],DH
0778:0029 D400 AAM 00
0778:002B 0000 ADD [BX+SI],AL
0778:002D 0000 ADD [BX+SI],AL
0778:002F 0000 ADD [BX+SI],AL
0778:0031 0000 ADD [BX+SI],AL
0778:0033 0000 ADD [BX+SI],AL
0778:0035 0000 ADD [BX+SI],AL
0778:0037 0000 ADD [BX+SI],AL
0778:0039 0C50 OR AL,50
0778:003B 41 INC CX
0778:003C 53 PUSH BX
0778:003D 53 PUSH BX
0778:003E 57 PUSH DI
0778:003F 4F DEC DI
```

(2) 在进行计算前，显示ID、BUFFER的内存值的截图（多显示、少显示均扣分）

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
0778:0026 42 INC DX
0778:0027 3030 XOR [BX+SI],DH
0778:0029 D400 AAM 00
0778:002B 0000 ADD [BX+SI],AL
0778:002D 0000 ADD [BX+SI],AL
0778:002F 0000 ADD [BX+SI],AL
0778:0031 0000 ADD [BX+SI],AL
0778:0033 0000 ADD [BX+SI],AL
0778:0035 0000 ADD [BX+SI],AL
0778:0037 0000 ADD [BX+SI],AL
0778:0039 0C50 OR AL,50
0778:003B 41 INC CX
0778:003C 53 PUSH BX
0778:003D 53 PUSH BX
0778:003E 57 PUSH DI
0778:003F 4F DEC DI
-g5
AX=0776 BX=0000 CX=014B DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0776 ES=0766 SS=0775 CS=0778 IP=0005 NU UP EI PL NZ NA PO NC
0778:0005 BB1400 MOV BX,0014
-d 0776:0000 0013
0776:0000 32 32 30 34 31 31 32 39-31 33 00 00 00 00 00 00 2204112913.....
0776:0010 00 00 00 00 .....
```

(3) 输入回车后，显示ID、BUFFER的内存值的截图（多显示、少显示均扣分）

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
0778:0026 42      INC      DX
0778:0027 3030     XOR      [BX+SI],DH
0778:0029 D400     AAM      00
0778:002B 0000     ADD      [BX+SI],AL
0778:002D 0000     ADD      [BX+SI],AL
0778:002F 0000     ADD      [BX+SI],AL
0778:0031 0000     ADD      [BX+SI],AL
0778:0033 0000     ADD      [BX+SI],AL
0778:0035 0000     ADD      [BX+SI],AL
0778:0037 0000     ADD      [BX+SI],AL
0778:0039 0C50     OR       AL,50
0778:003B 41      INC      CX
0778:003C 53      PUSH     BX
0778:003D 53      PUSH     BX
0778:003E 57      PUSH     DI
0778:003F 4F      DEC      DI
-g23
2204112913
AX=4C00 BX=0014 CX=0000 DX=0000 SP=0000 BP=0000 SI=000A DI=0000
DS=0776 ES=0766 SS=0775 CS=0778 IP=0023  NU UP EI PL NZ NA PE NC
0778:0023 CD21      INT      21
-d 0776:0000 0013
0776:0000 32 32 30 34 31 31 32 39-31 33 09 09 07 03 05 05  2204112913.....
0776:0010 09 04 05 01                                     ....
```

(4) 源代码

```
1  name Password
2  title Set Password
3
4  assume cs:code, ds:data
5
6  data segment
7      ID      db '2204112913'
8      BUFFER  db 10 dup(?)
9      TABLE  db 7, 5, 9, 1, 3, 6, 8, 0, 2, 4
10 data ends
11
12 code segment
13
14     main proc far
15         mov ax, seg data
16         mov ds, ax
17         mov bx, offset TABLE
18         mov si, 0
19
20         mov cx, 10
21     read_loop:
22         mov ah, 01h
23         int 21h
24         cmp al, 0Dh
25
26         je read_done
```

```
26
27     sub al, '0'
28     xlat
29     mov BUFFER[si], al
30     inc si
31     loop read_loop
32 read_done:
33
34     mov ax, 4c00h
35     int 21h
36
37     main endp
38
39 code ends
40     end main
```