

ОТЧЕТ

По лабораторной работе №1

По курсу: «Операционные системы»

Тема: «Прерывание таймера INT 8h и его функции»

Студент: Керимов А. Ш.

Группа: ИУ7-54Б

Преподаватель: Рязанова Н. Ю.

Листинг обработчика INT 8h

```
1 ;; Вызов процедуры sub_1
2 020A:0746 E8 0070          call    sub_1          ; (07B9)
3
4 ;; Сохранение регистров ES, DS, AX, DX
5 020A:0749 06              push    es
6 020A:074A 1E              push    ds
7 020A:074B 50              push    ax
8 020A:074C 52              push    dx
9
10 ;; Установка сегмента данных
11 020A:074D B8 0040          mov     ax,40h
12 020A:0750 8E D8           mov     ds,ax
13 020A:0752 33 C0           xor     ax,ax          ; Zero register
14 020A:0754 8E C0           mov     es,ax
15
16 ;; Инкремент счётчика реального времени по известному адресу в области дан
   ных BIOS
17 ; Инкремент двух младших байтов счётчика реального времени по адресу
   0040:006C
18 020A:0756 FF 06 006C      inc     word ptr ds:[6Ch] ; (0040:006C=0E873h)
19 020A:075A 75 04           jnz     loc_1          ; Jump if not zero
20 ; Инкремент двух старших байтов счётчика реально времени по адресу
   0040:006E
21 020A:075C FF 06 006E      inc     word ptr ds:[6Eh] ; (0040:006E=0Ch)
22
23 ;; Сброс счётчика реального времени при наступлении новых суток
24 020A:0760          loc_1:
25 020A:0760 83 3E 006E 18     cmp     word ptr ds:[6Eh],18h ; (0040:006E=0Ch)
26 020A:0765 75 15           jne     loc_2          ; Jump if not equal
27 020A:0767 81 3E 006C 00B0   cmp     word ptr ds:[6Ch],0B0h ; (0040:006C=0E873h)
28 020A:076D 75 0D           jne     loc_2          ; Jump if not equal
29 ; Обнуление двух старших байтов счётчика реального времени по адресу
   0040:006E
30 020A:076F A3 006E           mov     word ptr ds:[6Eh],ax ; (0040:006E=0Ch)
31 ; Обнуление двух младших байтов счётчика реального времени по адресу
   0040:006C
32 020A:0772 A3 006C           mov     word ptr ds:[6Ch],ax ; (0040:006C=0E873h)
33 ; Установка флага прошедших суток по адресу 0040:0070
34 020A:0775 C6 06 0070 01     mov     byte ptr ds:[70h],1 ; (0040:0070=0)
35 020A:077A 0C 08           or      al,8
36
37 ;; Декремент счётчика времени до отключения моторчика дисководов по известн
   ому адресу в области данных BIOS
```

```

38 020A:077C          loc_2:
39 020A:077C  50          push    ax
40 020A:077D  FE 0E 0040        dec byte ptr ds:[40h]      ; (0040:0040=2Bh)
41 020A:0781  75 0B          jnz loc_3                ; Jump if not zero
42 ; Установка флага отключения моторчика дисковода
43 020A:0783  80 26 003F F0      and byte ptr ds:[3Fh],0F0h ; (0040:003F=0)
44 ; Посылание команды отключения 0Ch в порт дисковода 3F2h
45 020A:0788  B0 0C          mov al,0Ch
46 020A:078A  BA 03F2        mov dx,3F2h
47 020A:078D  EE          out dx,al      ; port 3F2h, dsk0 contrl output
48
49 ;; Вызов прерывания 1Ch
50 020A:078E          loc_3:
51 020A:078E  58          pop ax
52 ; Установлен ли флаг разрешения прерываний IF?
53 020A:078F  F7 06 0314 0004 test word ptr ds:[314h],4 ;
    (0040:0314=3200h)
54 020A:0795  75 0C          jnz loc_4                ; Jump if not zero
55 ; Косвенный вызов прерывания 1Ch
56 020A:0797  9F          lahf                    ; Load ah from flags
57 020A:0798  86 E0          xchg ah,al
58 020A:079A  50          push ax
59 020A:079B  26: FF 1E 0070    call dword ptr es:[70h] ; (0000:0070=6ADh)
60 020A:07A0  EB 03          jmp short loc_5          ; (07A5)
61 020A:07A2  90          nop
62 ; Вызов прерывания 1Ch
63 020A:07A3          loc_4:
64 020A:07A3  CD 1C          int 1Ch      ; Timer break (call each 18.2ms)
65
66 ;; Вызов процедуры sub_1
67 020A:07A5          loc_5:
68 020A:07A5  E8 0011        call sub_1      ; (07B9)
69
70 ;; Сброс контроллера прерываний записью 20h в порт 20h
71 020A:07A8  B0 20          mov al,20h      ; ' '
72 020A:07AA  E6 20          out 20h,al      ; port 20h, 8259-1 int command
73 ; al = 20h, end of interrupt
74
75 ;; Восстановление регистров DX, AX, DS, ES
76 020A:07AC  5A          pop dx
77 020A:07AD  58          pop ax
78 020A:07AE  1F          pop ds
79 020A:07AF  07          pop es
80
81 ;; Завершение обработчика прерывания 8h
82 020A:07B0  E9 FE99        jmp $-164h      ; (07B0-164=064C)
83
84 020A:064C  1E          push ds

```

```

85 020A:064D 50          push    ax
86 ; ...
87 020A:06AA 58          pop     ax
88 020A:06AB 1F          pop     ds
89 020A:06AC CF          ired           ; Interrupt return

```

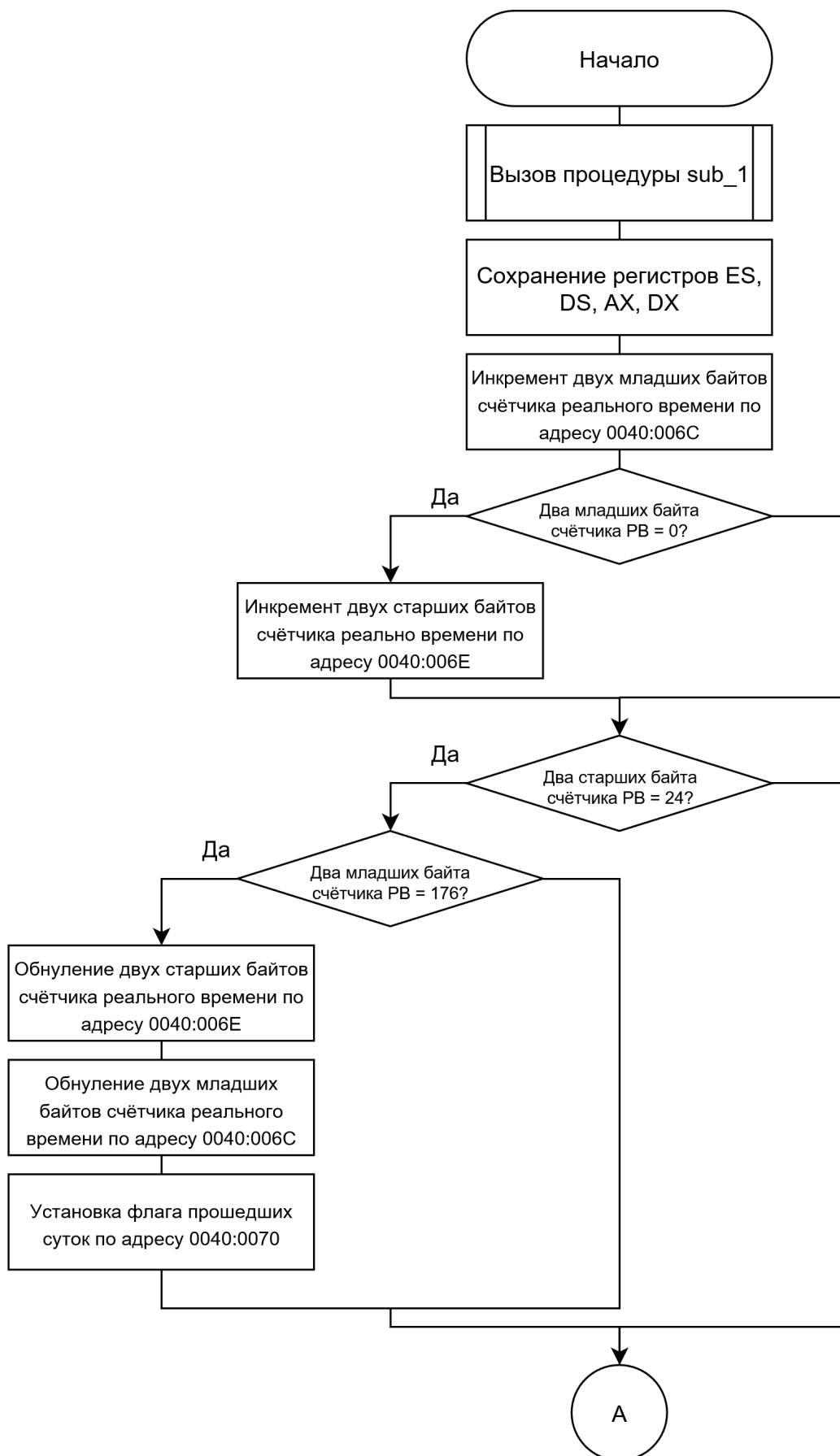
Листинг процедуры sub_1

```

1      sub_1      proc      near
2
3  ;; Сохранение регистров DS, AX
4  020A:07B9 1E          push    ds
5  020A:07BA 50          push    ax
6
7  ;; Установка сегмента данных = 0040
8  020A:07BB B8 0040      mov     ax,40h
9  020A:07BE 8E D8        mov     ds,ax
10
11 ; Загрузка EFLAGS в AH
12 020A:07C0 9F          lahf           ; Load ah from flags
13 020A:07C1 F7 06 0314 2400 test    word ptr ds:[314h],2400h ;
    (0040:0314=3200h)
14 020A:07C7 75 0C        jnz     loc_7           ; Jump if not zero
15 020A:07C9 F0> 81 26 0314 FDF lock    and word ptr ds:[314h],0FDFh ;
    (0040:0314=3200h)
16 020A:07D0          loc_6:
17 020A:07D0 9E          sahf           ; Store ah into flags
18 020A:07D1 58          pop     ax
19 020A:07D2 1F          pop     ds
20 020A:07D3 EB 03        jmp     short loc_8      ; (07D8)
21 020A:07D5          loc_7:
22 020A:07D5 FA          cli           ; Disable interrupts
23 020A:07D6 EB F8        jmp     short loc_6      ; (07D0)
24 020A:07D8          loc_8:
25 020A:07D8 C3          retn
26 sub_1      endp

```

Схема алгоритма обработчика INT 8h



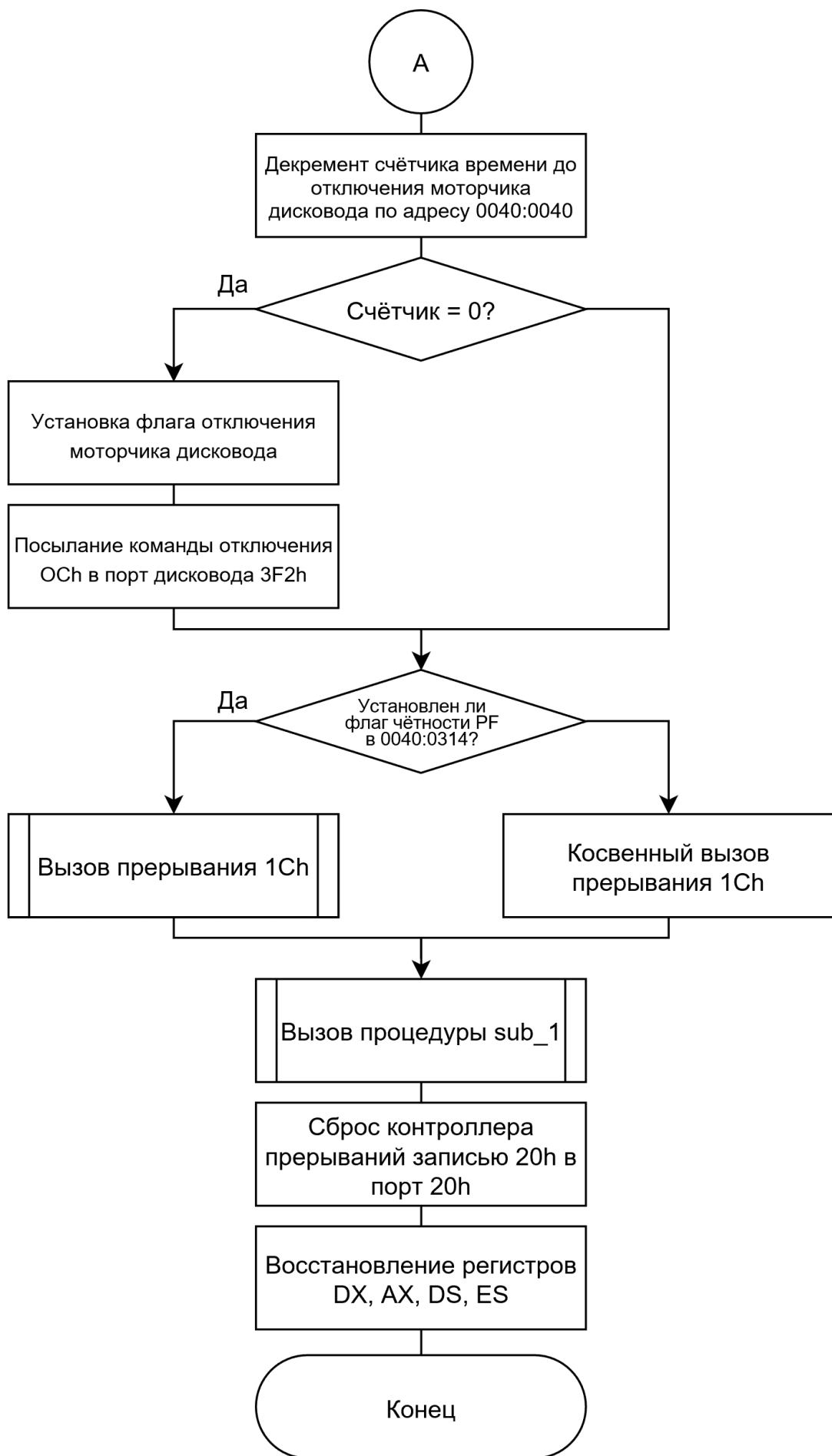


Схема алгоритма процедуры sub_1

