



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Московский государственный технический университет имени
Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Лабораторная работа №1 по дисциплине "Операционные системы"

Тема Дизассемблирование INT 8h

Студент Мансуров В. М.

Группа ИУ7-56Б

Преподаватель Рязанова Н.Ю.

1. Полученный дизассемблированный код

1.1. Листинг обработчика прерывания INT 8

```
1      Temp.lst                      Sourcer Listing v3.07      11-Sep-22      9:18 pm
2      Page 1
3      ;; Вызов подпрограммы sub_6:
4      020A:0746  E8 0070              call     sub_6      ; (07B9)
5
6      ;; Сохранение значений регистров es, ds, ax, dx:
7      020A:0749  06                  push     es
8      020A:074A  1E                  push     ds
9      020A:074B  50                  push     ax
10     020A:v074C  52                  push     dx
11
12     ;; Загрузка сегментных регистров ds, es:
13     ;; (40h - сегментная часть адреса области данных BIOS)
14     020A:074D  B8 0040              mov     ax,40h
15     020A:0750  8E D8              mov     ds,ax
16     020A:0752  33 C0              xor     ax,ax
17     020A:0754  8E C0              mov     es,ax
18
19     ;; Инкремент счётчиков таймера:
20     ;; 0040:006C, 0040:006E - адреса младшего и старшего слова
21     ;; счётчика прерываний таймера BIOSa
22     020A:0756  FF 06 006C          inc     word ptr ds:[6Ch]
23     ;; (0040:006C=4E47h), по этому адресу располагается счетчик реального времени
24     020A:075A  75 04              jnz     loc_3      ; Jump if not zero
25     020A:075C  FF 06 006E          inc     word ptr ds:[6Eh]      ; (0040:006E=15h)
26
27     ;; Сброс счётчиков времени при наступлении нового дня:
28     ;; 0040:006E == 18h (24), 0040:006C == B0h (176)
29     ;; 18h << 16 + B0h == 24 * 60 * 60 * c;
30     ;; c = 1573040 / 86400 = 18.2... - количество срабатываний таймера в секунду
31     ;; Таким образом из того, что условие выполняется, следует, что прошли сутки.
32     020A:0760                      loc_3:
33     020A:0760  83 3E 006E 18        cmp     word ptr ds:[6Eh],18h      ; (0040:006E=15h)
34     020A:0765  75 15              jne     loc_4      ; Jump if not equal
35     020A:0767  81 3E 006C 00B0      cmp     word ptr ds:[6Ch],0B0h     ; (0040:006C=4E47h)
36     020A:076D  75 0D              jne     loc_4      ; Jump if not equal
37
38     ;; Обнуление счетчика (старшего слова и младшего слова) таймера
39     020A:076F  A3 006E              mov     word ptr ds:[6Eh],ax      ; (0040:006E=15h)
40     020A:0772  A3 006C              mov     word ptr ds:[6Ch],ax      ; (0040:006C=4E47h)
41
42     ;; Прошло более 24 часов, занесение значения 1 в 0040:0070
43     020A:0775  C6 06 0070 01        mov     byte ptr ds:[70h],1      ; (0040:0070=0)
44     ;; Установка al = 8:
45     020A:077A  0C 08              or      al,8
46
47     020A:077C                      loc_4:
48     ;; Сохранение регистра ax:
49     020A:077C  50                  push     ax
50     ;; Декремент счётчика времени до отключения моторчика дисковод:
51     ;; (0040:0040 - адрес счётчика времени в области данных накопителя FDD)
52     020A:077D  FE 0E 0040          dec     byte ptr ds:[40h]      ; (0040:0040=96h)
53     020A:0781  75 0B              jnz     loc_5      ; Jump if not zero
54     ;; Установка флагов, отвечающих за отключение моторчика дисковод:
55     020A:0783  80 26 003F F0          and     byte ptr ds:[3Fh],0F0h   ; (0040:003F=0)
56     ;; Отправка команды отключения моторчика дисковод:
57     020A:0788  B0 0C              mov     al,0Ch
58     020A:078A  BA 03F2          mov     dx,3F2h
59     020A:078D  EE                      out     dx,al      ; port 3F2h, dsk0 contrl output
60
61     020A:078E                      loc_5:
62     ;; Восстановление регистра ax:
63     020A:078E  58                  pop      ax
64     ;; Проверка второго бита (Parity Flag - флаг чётности):
65     ;; 0040:0314h - адрес области данных BIOS, содержащей копию флагов
66     020A:078F  F7 06 0314 0004      test     word ptr ds:[314h],4      ; (0040:0314=3200h)
67     020A:0795  75 0C              jnz     loc_6      ; Jump if not zero
68     ;; Сохранение младшего байта регистра FLAGS в AH:
69     020A:0797  9F                      lahf                      ; Load ah from flags
70     ;; Обмен значений регистров ah и al:
71     ;; Теперь младший байт регистра FLAGS находится в младшем байте регистра ax
```

```

72 020A:0798 86 E0          xchg    ah,al
73 ;; Сохранение регистра ax:
74 020A:079A 50            push    ax
75 ;; Косвенный вызов пользовательского прерывания по адресу в таблице векторов прерываний:
76 ;; В этом случае не произойдёт push регистра FLAGS, на его месте будет AX,
77 ;; который восстановится в регистр FLAGS после выхода из обработчика прерывания
78 020A:079B 26: FF 1E 0070    call    dword ptr es:[70h]      ; (0000:0070=6ADh)
79 020A:07A0 EB 03          jmp     short loc_7             ; (07A5)
80 020A:07A2 90            nop
81 ;; Вызов пользовательского прерывания через int 1Ch:
82 020A:07A3          loc_6:
83 020A:07A3 CD 1C          int     1Ch                   ; Timer break (call each 18.2ms)
84 ;; Вызов подпрограммы sub_6:
85 020A:07A5          loc_7:
86 020A:07A5 E8 0011        call    sub_6                 ; (07B9)
87 ;; Сброс контроллера прерываний (отправка команды End Of Interrupt):
88 ;; Разрешение обработки прерываний с текущим или более низким приоритетом
89 020A:07A8 B0 20          mov     al,20h                ; ' '
90 020A:07AA E6 20          out     20h,al              ; port 20h, 8259-1 int command
91                                     ; al = 20h, end of interrupt
92 ;; Восстановление значений регистров es, ds, ax, dx:
93 020A:07AC 5A            pop     dx
94 020A:07AD 58            pop     ax
95 020A:07AE 1F            pop     ds
96 020A:07AF 07            pop     es
97
98 020A:07B0 E9 FE99        jmp     $-164h
99 020A:07B3 C4            db     0C4h
100 020A:07B4 C4 0E 93E9    les     cx,dword ptr ds:[93E9h] ; (0000:93E9=3C0Ch) Load 32 bit
    ptr
101 020A:07B8 FE            db     0FEh

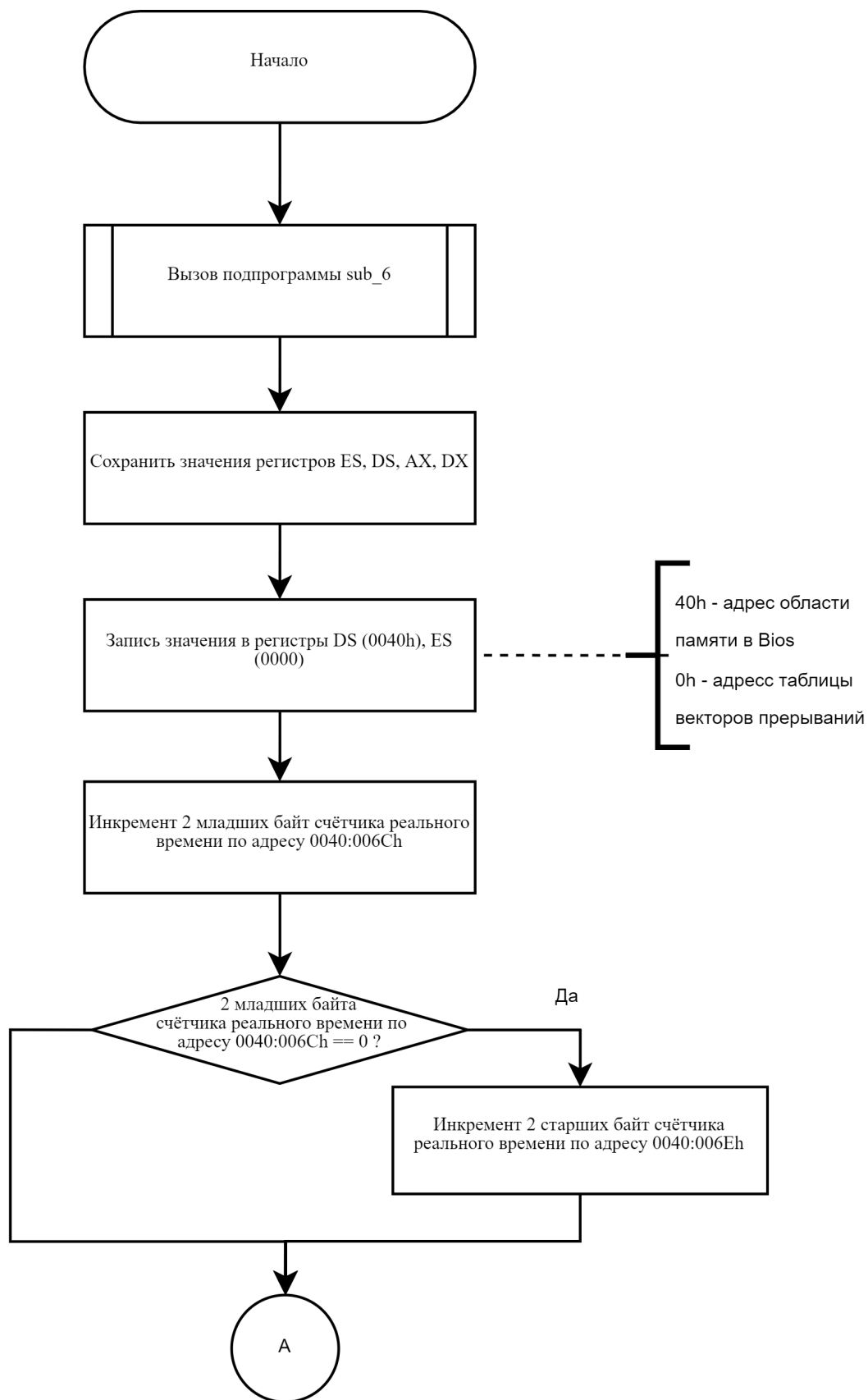
```

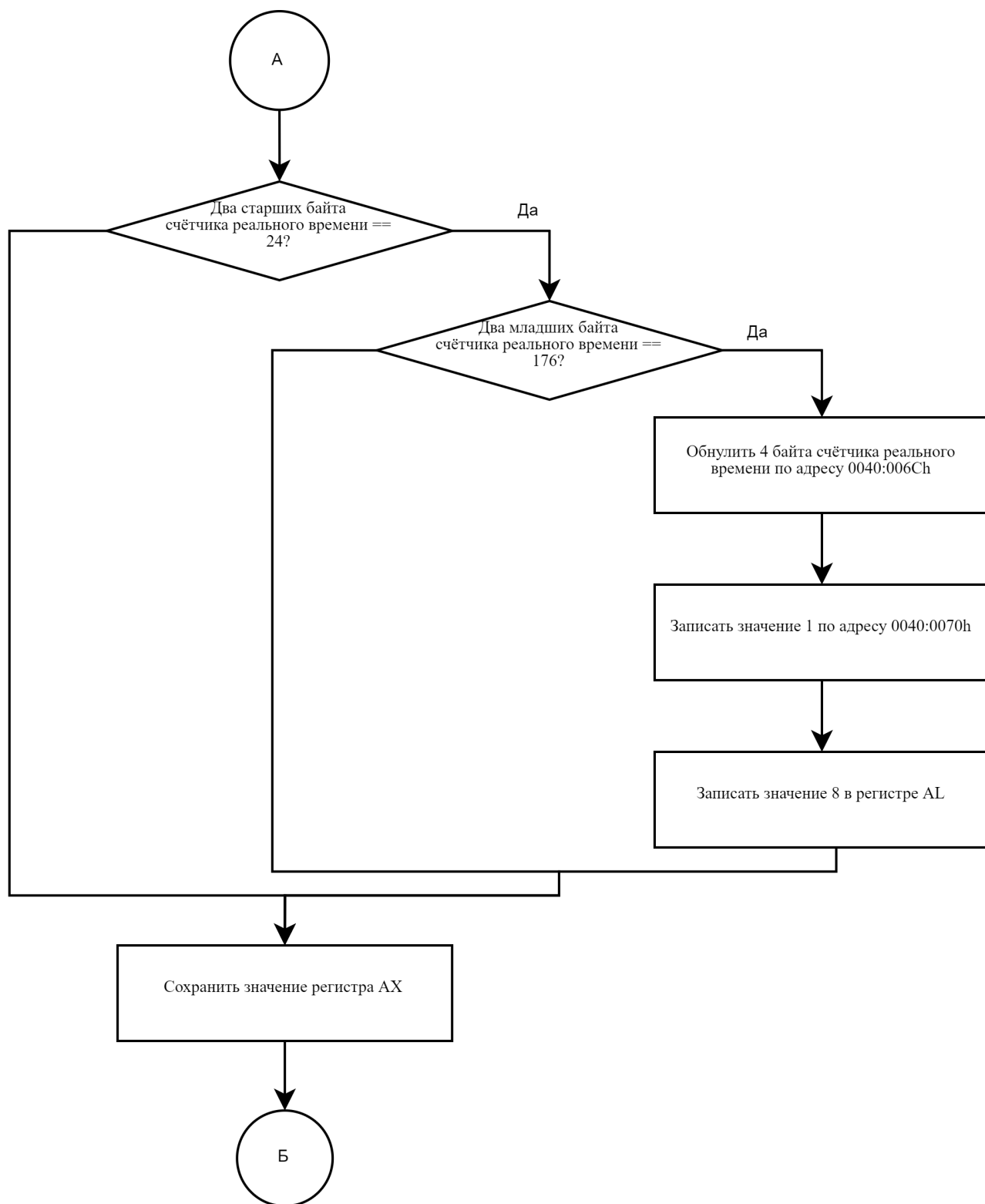
1.2. Листинг процедуры sub_6

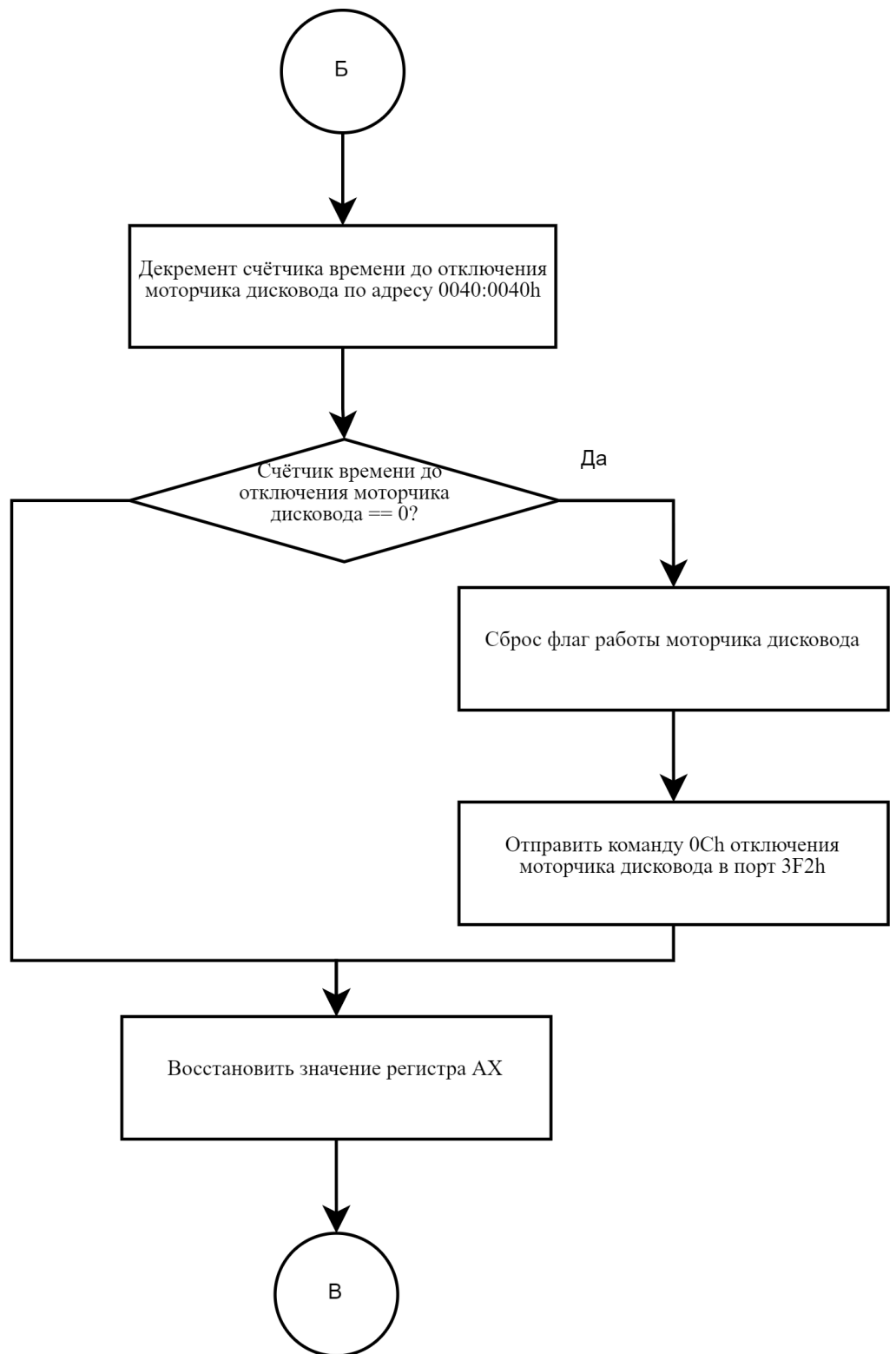
```
1 ; SUBROUTINE
2 Temp.lst Sourcer Listing v3.07 11-Sep-22 9:18 pm Page 2
3 sub_6 proc near
4 ;; Сохранение значений регистров ds, ax:
5 020A:07B9 1E push ds
6 020A:07BA 50 push ax
7 ;; Загрузка сегментного регистра ds:
8 020A:07BB B8 0040 mov ax,40h
9 020A:07BE 8E D8 mov ds,ax
10 ;; Сохранение младшего байта регистра FLAGS в AH:
11 020A:07C0 9F lahf ; Load ah from flags
12 ;; Проверка DF и старшего бита IOPL по адресу 0040:0314h:
13 020A:07C1 F7 06 0314 2400 test word ptr ds:[314h],2400h ; (0040:0314=3200h)
14 020A:07C7 75 0C jnz loc_9 ; Jump if not zero
15 ;; Обнуление 9 бита - сброс IF (запрет прерываний):
16 020A:07C9 F0> 81 26 0314 FDFF lock and word ptr ds:[314h],0FDFFh ; (0040:0314=3200h)
17
18 020A:07D0 loc_8:
19 ;; Сохранение регистра AH в младший байт FLAGS:
20 020A:07D0 9E sahf ; Store ah into flags
21 ;; Восстановление значений регистров ds, ax:
22 020A:07D1 58 pop ax
23 020A:07D2 1F pop ds
24
25 020A:07D3 EB 03 jmp short loc_10 ; (07D8)
26
27 020A:07D5 loc_9:
28 ;; Сброс IF ( Interrupt flag )
29 020A:07D5 FA cli ; Disable interrupts
30 020A:07D6 EB F8 jmp short loc_8 ; (07D0)
31
32 020A:07D8 loc_10:
33 ;; Возврат из подпрограммы:
34 020A:07D8 C3 retn
35 sub_6 endp
```

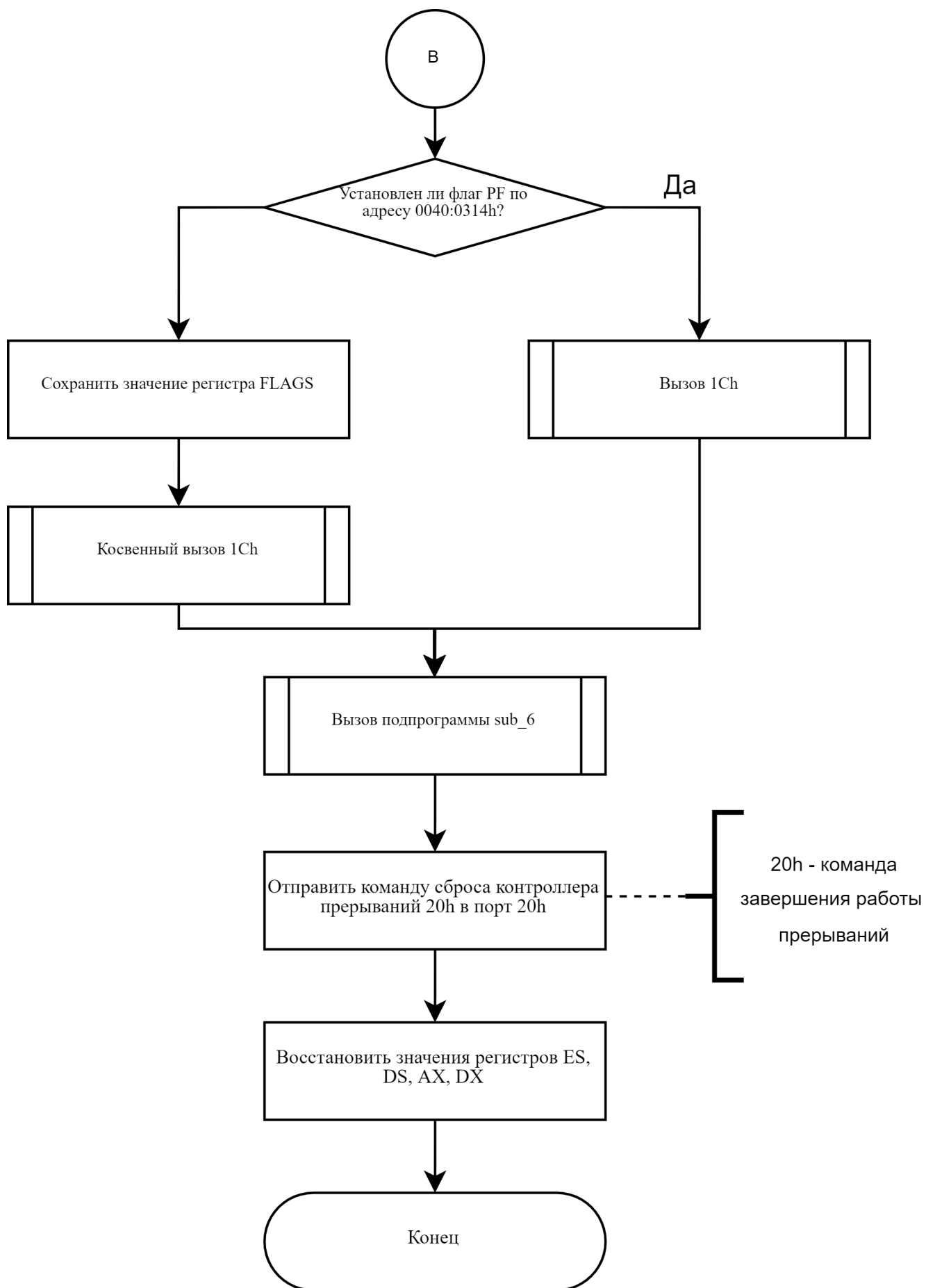
2. Схема алгоритмов

2.1. Схема алгоритма обработчика INT8h









2.2. Схема алгоритма процедуры sub_6

