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 (часть 1) по дисциплине "Операционные системы"

Тема Дизассемблирование INT 8h

Студент Смирнов И.В.

Группа ИУ7-52Б

Преподаватель Рязанова Н. Ю.

Москва — 2024 г.

1. Полученный дизассемблированный код

1.1. Листинг обработчика прерывания INT 8h

```
1  ; Вызов субротины sub_1
2  020A:0746  E8 0070          call    sub_1           ; (07B9)
3  ; Сохранение ES, DS, AX, DX
4  020A:0749  06               push    es
5  020A:074A  1E               push    ds
6  020A:074B  50               push    ax
7  020A:074C  52               push    dx
8  ; Поместить 0040h в DS (адрес начала области данных BIOS)
9  020A:074D  B8 0040          mov ax,40h
10 020A:0750  8E D8            mov ds,ax
11 ; Поместить 0 в ES (адресом начала таблица векторов прерываний)
12 020A:0752  33 C0            xor ax,ax           ; Zero register
13 020A:0754  8E C0            mov es,ax
14 ; Инкремент младшего слова счетчика реального времени
15 020A:0756  FF 06 006C       inc word ptr ds:[6Ch]   ; (0040:006C=8E8Bh)
16 ; Если ZF=0 (счетчик не переполнился), то переход на loc_1
17 020A:075A  75 04            jnz loc_1           ; Jump if not zero
18 ; Инкремент часов 0040:006Eh (старшего слова счетчика суточного времени)
19 020A:075C  FF 06 006E       inc word ptr ds:[6Eh]   ; (0040:006E=14h)
20 020A:0760        loc_1:
21 ; Проверка, прошли ли сутки с момента запуска счетчика реального времени:
22 ; Если хотя бы одно из условий не выполняется, переход на loc_2
23 020A:0760  83 3E 006E 18       cmp word ptr ds:[6Eh],18h   ; (0040:006E=14h)
24 020A:0765  75 15            jne loc_2           ; Jump if not equal
25 020A:0767  81 3E 006C 00B0  cmp word ptr ds:[6Ch],0B0h  ; (0040:006C=8E8Bh)
26 020A:076D  75 0D            jne loc_2           ; Jump if not equal
27 ; Прошли очередные сутки с момента запуска таймера:
28 ; 1) сбросить счетчик реального времени
29 020A:076F  A3 006E          mov word ptr ds:[6Eh],ax    ; (0040:006E=14h)
30 020A:0772  A3 006C          mov word ptr ds:[6Ch],ax    ; (0040:006C=8E8Bh)
31 ; 2) поместить единицу в ячейку 0040:0070h
32 020A:0775  C6 06 0070 01       mov byte ptr ds:[70h],1 ; (0040:0070=0)
33 ; 3) загрузить значение 8 в AX (AX до этого был равен нулю)
34 020A:077A  0C 08            or  al,8
35 ; Декремент счётчика времени до отключения моторчика дисковода по известному адр
       есу в области данных BIOS
36 020A:077C        loc_2:
37 ; Сохранение значения регистра AX
38 020A:077C  50               push    ax
39 ; Декремент времени, оставшегося до выключения моторчика дисковода
40 020A:077D  FE 0E 0040       dec byte ptr ds:[40h]   ; (0040:0040=78h)
41 020A:0781  75 0B            jnz loc_3           ; Jump if not zero
42 ; 1) сброс соответствующих флагов моторчика дисковода (младшие 4 бита)
43 020A:0783  80 26 003F F0       and byte ptr ds:[3Fh],0F0h  ; (0040:003F=0)
44 ; 2) Загрузка в AL (данные для вывода в последующей команде out) значения
45 ; 0Ch=00001100b: 2 бит=1 (разрешение работы контроллера), 3 бит=1 (разрешение
46 ; прерываний и прямого доступа к памяти), 4?7 биты сброшены (значения 1 в каждом
47 ; разряде вызывали бы включение соответствующего двигателя НГМД)
48 020A:0788  B0 0C             mov al,0Ch
49 ; 3) Загрузка в DX (примемник в последующей команде out) номер порта 3F2
50 ; (порт цифрового управления)
51 020A:078A  BA 03F2          mov dx,3F2h
52 ; 4) вывод данных в порт
53 020A:078D  EE               out dx,al           ; port 3F2h, dsk0 contrl output
```

```
54 ; Вызов прерывания INT 1Ch
55 020A:078E        loc_3:
56 ; Восстановление значения AX
57 020A:078E  58            pop ax
58 ; Проверка флага четноси PF(0100 ? 2 бит в области BIOS по адресу
59 ; 0040:0314h, где находится копия флагов, отвечает за флаг PF)
60 020A:078F  F7 06 0314 0004  test    word ptr ds:[314h],4     ; (0040:0314=3200h)
61 020A:0795  75 0C            jnz loc_4              ; Jump if not zero
62 ; Иначе будет осуществлен косвенный вызов прерывания 1CH с другими флагами
63 ; Загрузка младшего байта FLAGS в регистр AH
64 020A:0797  9F            lahf                    ; Load ah from flags
65 ; Обмен AH (младший байт FLAGS) и AL (8) -> AX=[08][младший байт FLAGS]
66 020A:0798  86 E0         xchg    ah,al
67 ; Сохранение значения регистра AX
68 020A:079A  50            push    ax
69 ; Косвенный вызов прерывания 1Ch с помощью адреса в таблице векторов прерываний
70 ; (1Ch*4=28*4=112=70H)
71 020A:079B  26: FF 1E 0070   call    dword ptr es:[70h]   ; (0000:0070=6ADh)
72 ; Переход на loc_5
73 020A:07A0  EB 03            jmp short loc_5       ; (07A5)
74 020A:07A2  90            nop
75 ; Вызов прерывания 1CH
76 020A:07A3        loc_4:
77 020A:07A3  CD 1C            int 1Ch             ; Timer break (call each 18.2ms)
78 ; Вызов подпрограммы sub_1 (запрет прерываний)
79 020A:07A5        loc_5:
80 020A:07A5  E8 0011          call    sub_1             ; (07B9)
81 ; Сброс контроллера прерываний
82 ; (Чтобы позволить прерываниям меньшего приоритета обрабатываться)
83 020A:07A8  B0 20            mov al,20h              ; ' '
84 020A:07AA  E6 20            out 20h,al              ; port 20h, 8259-1 int command
85                                                     ;  al = 20h, end of interrupt
86 ; Восстановление значений регистров DX, AX, DS, ES
87 020A:07AC  5A            pop dx
88 020A:07AD  58            pop ax
89 020A:07AE  1F            pop ds
90 020A:07AF  07            pop es
91 ; Переход в сторону выхода (020A:07B0  - 164h = 020A:064C)
92 020A:07B0  E9 FE99          jmp $-164h
93 ; Б
94 ; Сохраненение DS, AX
95 020A:064C  1E            push    ds
96 020A:064D  50            push    ax
97 ; Б
98 ; Восстановление DS, AX
99 020A:06AA  58            pop ax
100 020A:06AB  1F           pop ds
101 ; Выход из прерывания
102 020A:06AC  CF           iret                    ; Interrupt return
```

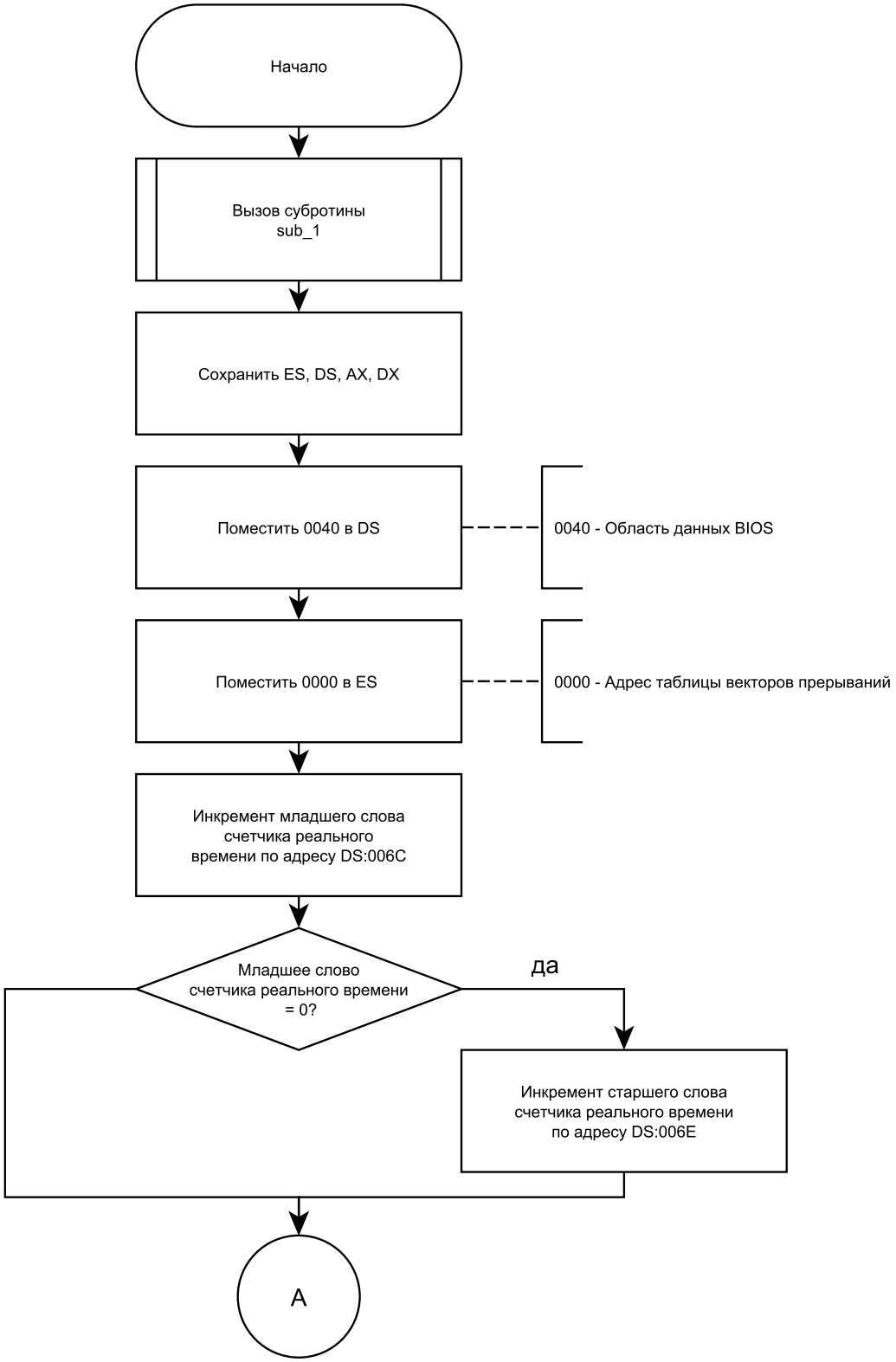
1.2. Листинг субротины sub_1

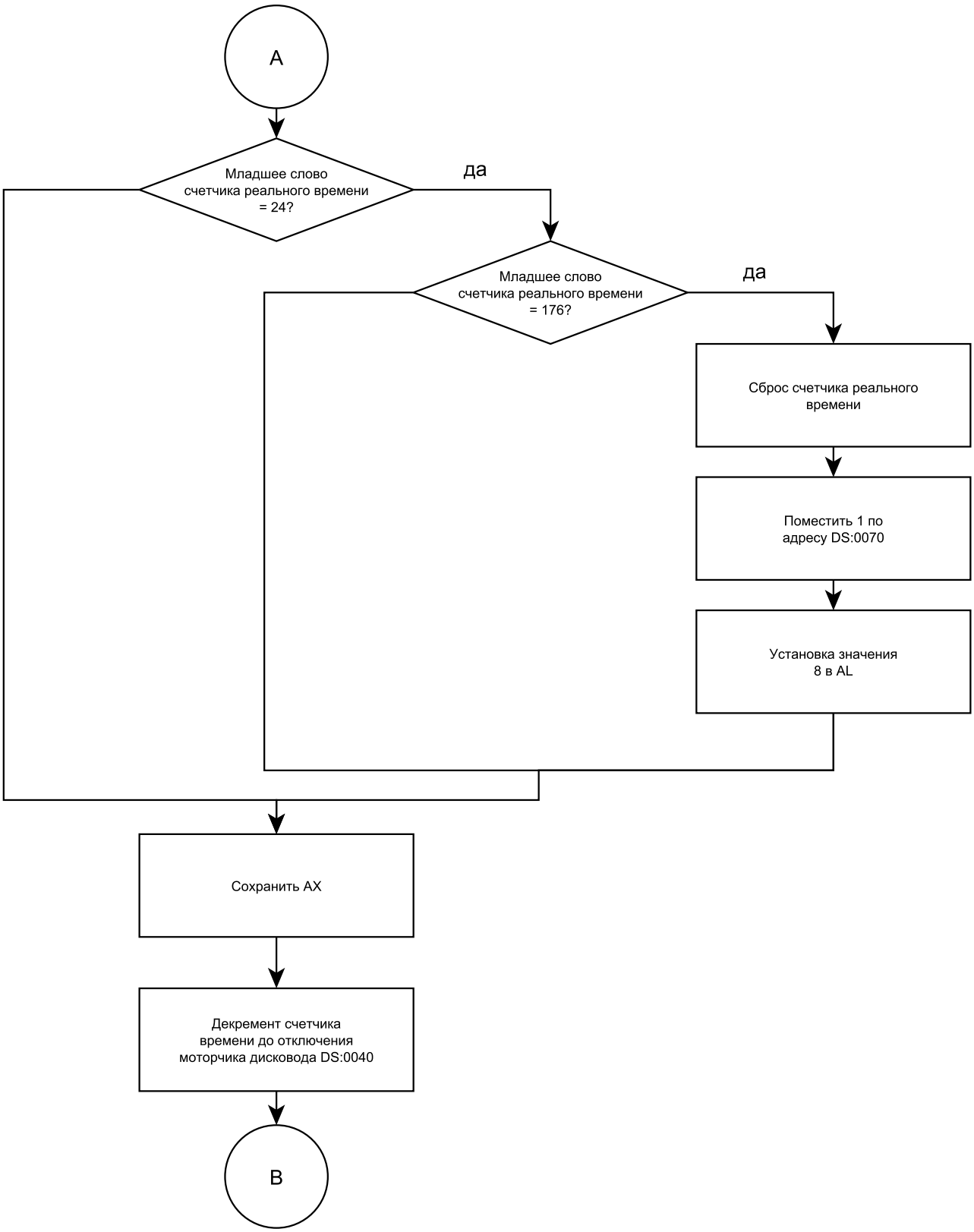
```
1              sub_1        proc    near
2 ; Сохранение DS, AX
3 020A:07B9  1E             push    ds
4 020A:07BA  50             push    ax
```

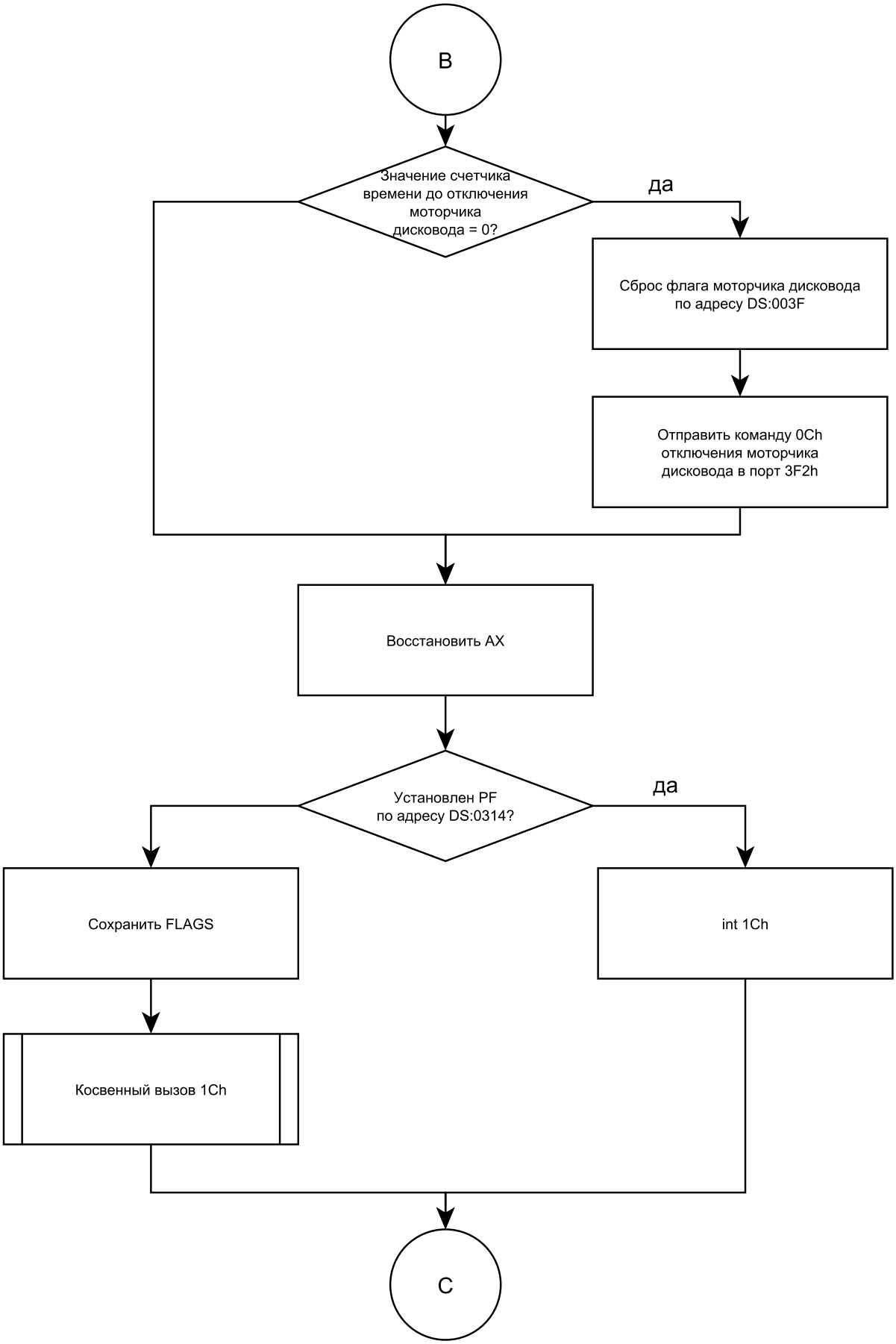
```
5  ; Поместить 0040h в DS (адрес начала области данных BIOS)
6  020A:07BB  B8 0040           mov ax,40h
7  020A:07BE  8E D8             mov ds,ax
8  ; Загрузка младшего байта FLAGS в регистр AH
9  020A:07C0  9F                lahf                    ; Load ah from flags
10 ; Проверка: поднят хотя бы один из флагов 10 или 13
11 020A:07C1  F7 06 0314 2400   test    word ptr ds:[314h],2400h     ;
       (0040:0314=3200h)
12 ; Если поднят хотя бы один, то переход на loc_22, чтобы командой cli сбросить
13 020A:07C7  75 0C             jnz loc_22              ; Jump if not zero
14 020A:07C9  F0> 81 26 0314 FDFF    lock and  word ptr ds:[314h],0FDFFh    ;
       (0040:0314=3200h)
15 020A:07D0        loc_21:
16 ; Восстановление SF, ZF, AF, PF и CF регистра FLAGS из AH
17 020A:07D0  9E                sahf                    ; Store ah into flags
18 ; Восстановление AX, DS
19 020A:07D1  58                pop ax
20 020A:07D2  1F                pop ds
21 ; Переход на loc_23
22 020A:07D3  EB 03             jmp short loc_23        ; (07D8)
23 020A:07D5        loc_22:
24 ; Сброс IF
25 020A:07D5  FA                cli                     ; Disable interrupts
26 020A:07D6  EB F8             jmp short loc_21        ; (07D0)
27 ; Выход из подпрограммы
28 020A:07D8        loc_23:
29 020A:07D8  C3                retn
30              sub_1           endp
```

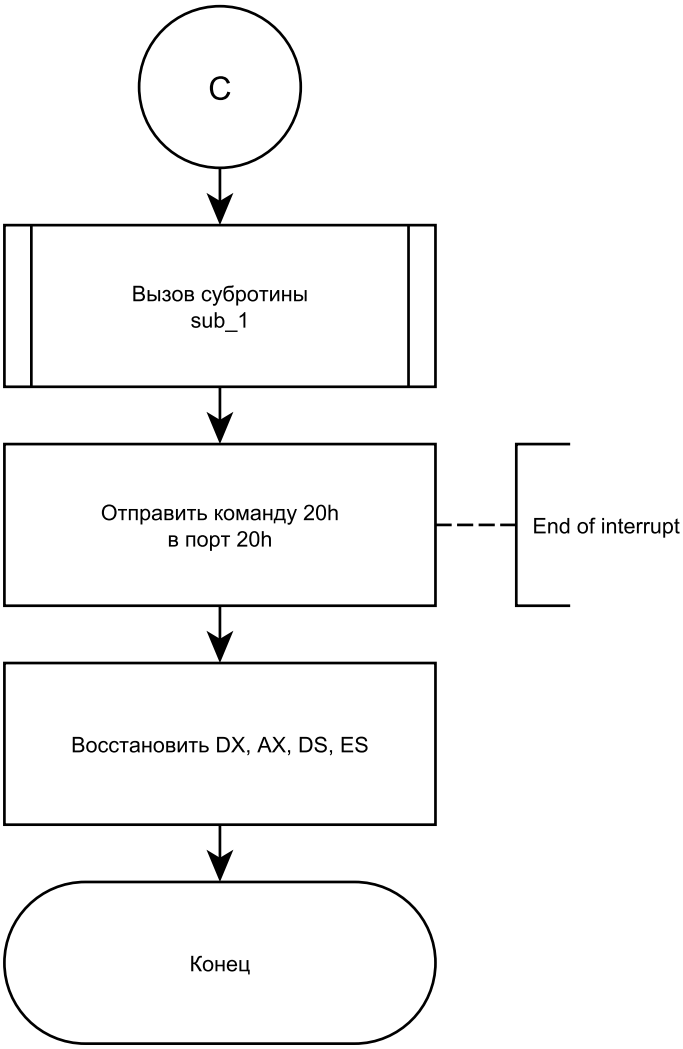
2. Схема алгоритмов

2.1. Схема алгоритма обработчика INT8h




A
Младшее слово
счетчика реального времени
= 24?
да
Младшее слово
счетчика реального времени
= 176?
да
Сброс счетчика реального
времени
Поместить 1 по
адресу DS:0070
Установка значения
8 в AL
Сохранить AX
Декремент счетчика
времени до отключения
моторчика дисковода DS:0040
B


B
Значение счетчика времени до отключения моторчика дисковода = 0?
да
Сброс флага моторчика дисковода по адресу DS:003F
Отправить команду 0Ch отключения моторчика дисковода в порт 3F2h
Восстановить AX
Установлен PF по адресу DS:0314?
да
Сохранить FLAGS
int 1Ch
Косвенный вызов 1Ch
C


C
Вызов субротины
sub_1
Отправить команду 20h
в порт 20h
End of interrupt
Восстановить DX, AX, DS, ES
Конец

2.2. Схема алгоритма процедуры sub_1