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»

Отчет по лабораторной работе №1 по курсу "Операционные системы"

Тема Дизассемблирование прерывания INT 8H

Студент Шацкий Р.Е.

Группа ИУ7-55Б

Оценка (баллы) _____

Преподаватели Рязанова Н.Ю.

Москва — 2021 г.

Получение дизассемблированного кода обработчика прерывания int 8h

Для выполнения лабораторной работы на виртуальную машину была поставлена операционная система Windows XP (32 бит).

Для определения адреса вектора из таблицы векторов прерываний нужно вычислить смещение. Так как номер прерывания — 8h, а длина far-адреса составляет 4, нужно умножить номер вектора на 4 и перевести полученное значение в шестнадцатеричную систему.

Получившееся значение – **20h**.

Для получения содержимого по адресу 0000:0020h, то есть адреса обработчика прерывания, используется программа-отладчик **AFDPRO**. Перейдя к адресу 0000:0020h, можно увидеть значения четырёх байт: **46 07 0A 02**.

Так как у байтов обратный порядок следования (little endian), нужно поменять порядок местами. Итоговый начальный адрес обработчика прерывания int 8h — **020A:0746**.

Получение дизассемблированного кода производится чс помощью утилиты sourcer. Для получения листинга кода нужно задать начальный и конечный адреса. Конец обработчика прерывания можно найти, зная, что код обработчика заканчивается командой **iret**. По адресу 020A:07B0 находится команда jmp $-164h. По смещению -164h находится несколько команд, в числе которых iret по адресу **020A:06AC**. Поэтому листинг кода выполнялся в два этапа: сначала получения кода от смещения 0746h до смещения 07B0h, а затем - от 064Ch до 06ACh.

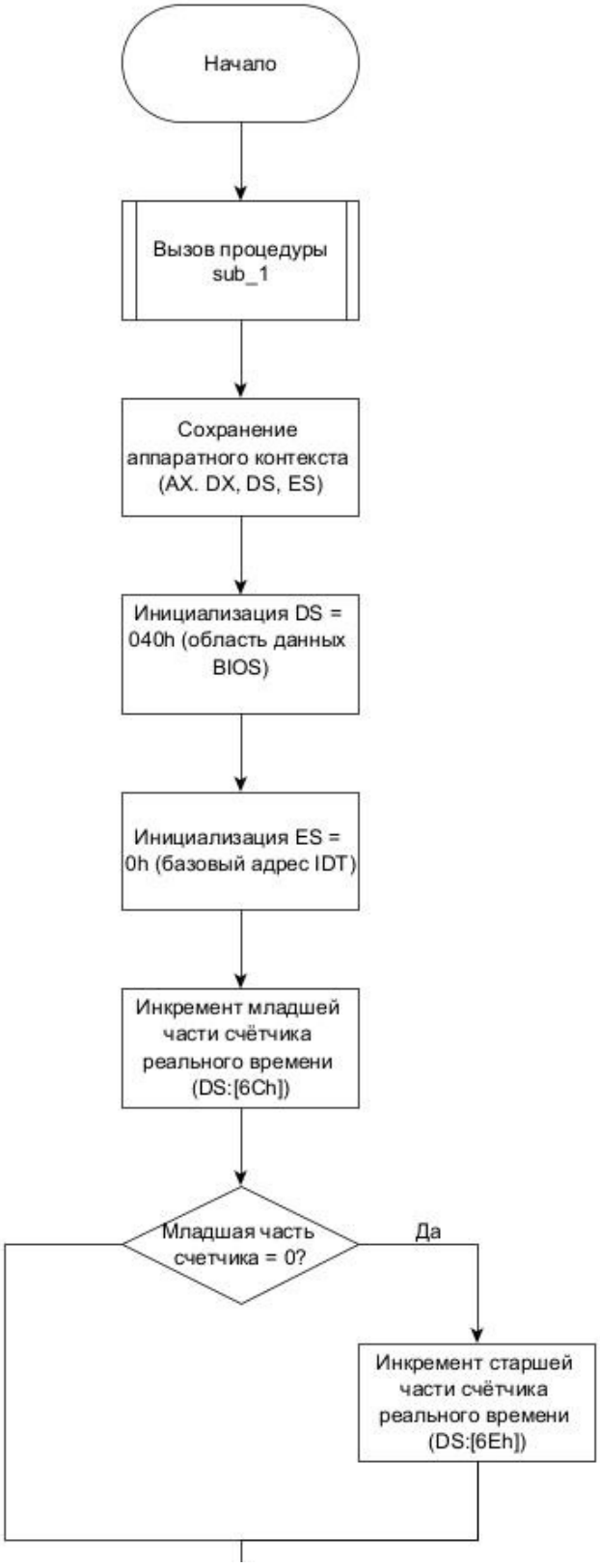
Листинг обработчика INT 8h

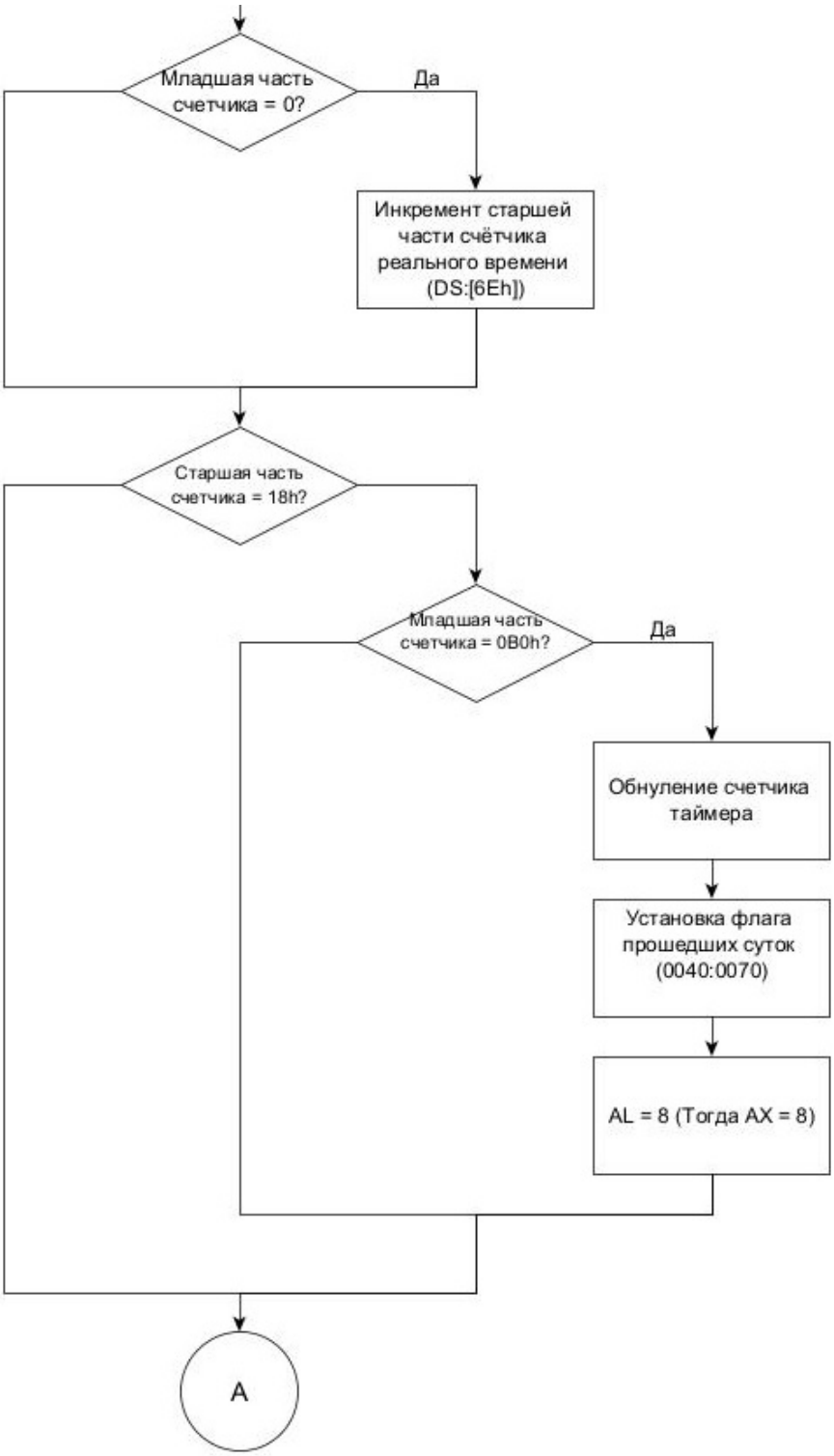
```
; Вызов subroutine1
020A:0746  E8 0070              call    sub_1           ; (07B9)
; Сохранение аппаратного контекста (es, ds, ax, dx)
020A:0749  06                   push    es
020A:074A  1E                   push    ds
020A:074B  50                   push    ax
020A:074C  52                   push    dx
; Установка 40h в DS, 0 в ES
020A:074D  B8 0040              mov ax,40h
020A:0750  8E D8                mov ds,ax
020A:0752  33 C0                xor ax,ax
020A:0754  8E C0                mov es,ax
; Инкремент младшего байта счетчика таймера
020A:0756  FF 06 006C               inc word ptr ds:[6Ch]   ; (0040:006C=82
    Fh)
; Инкремент старшего байта счетчика таймера, если младший занулился (FF ->
    00)
020A:075A  75 04                jnz loc_1           ; Jump if not zero
020A:075C  FF 06 006E               inc word ptr ds:[6Eh]   ; (0040:006E=12h
    )

; Проверка, прошло ли 24 часа
020A:0760           loc_1:
    ;  Если прошло 24 часа, то состояние счетчика - 1800B0, что равно
       1573040 = 18.2 * 60*60*24
    020A:0760  83 3E 006E 18            cmp word ptr ds:[6Eh],18h   ;
       (0040:006E=12h)
    020A:0765  75 15                jne loc_2           ; Jump if not equal
    020A:0767  81 3E 006C 00B0          cmp word ptr ds:[6Ch],0B0h  ;
       (0040:006C=82Fh)
    020A:076D  75 0D                jne loc_2           ; Jump if not equal
    ; Занулить счетчик таймера
    020A:076F  A3 006E              mov word ptr ds:[6Eh],ax    ; (0040:006E
       =12h)
    020A:0772  A3 006C              mov word ptr ds:[6Ch],ax    ; (0040:006C
       =82Fh)
    ; Установка флага прошедших суток по адресу 0000:0470, если прошло 24 ча
       са
    020A:0775  C6 06 0070 01            mov byte ptr ds:[70h],1 ;
       (0040:0070=0)
    020A:077A  0C 08                or  al,8

; Работа с моторчиком дисковода
020A:077C           loc_2:
    020A:077C  50                   push    ax
```

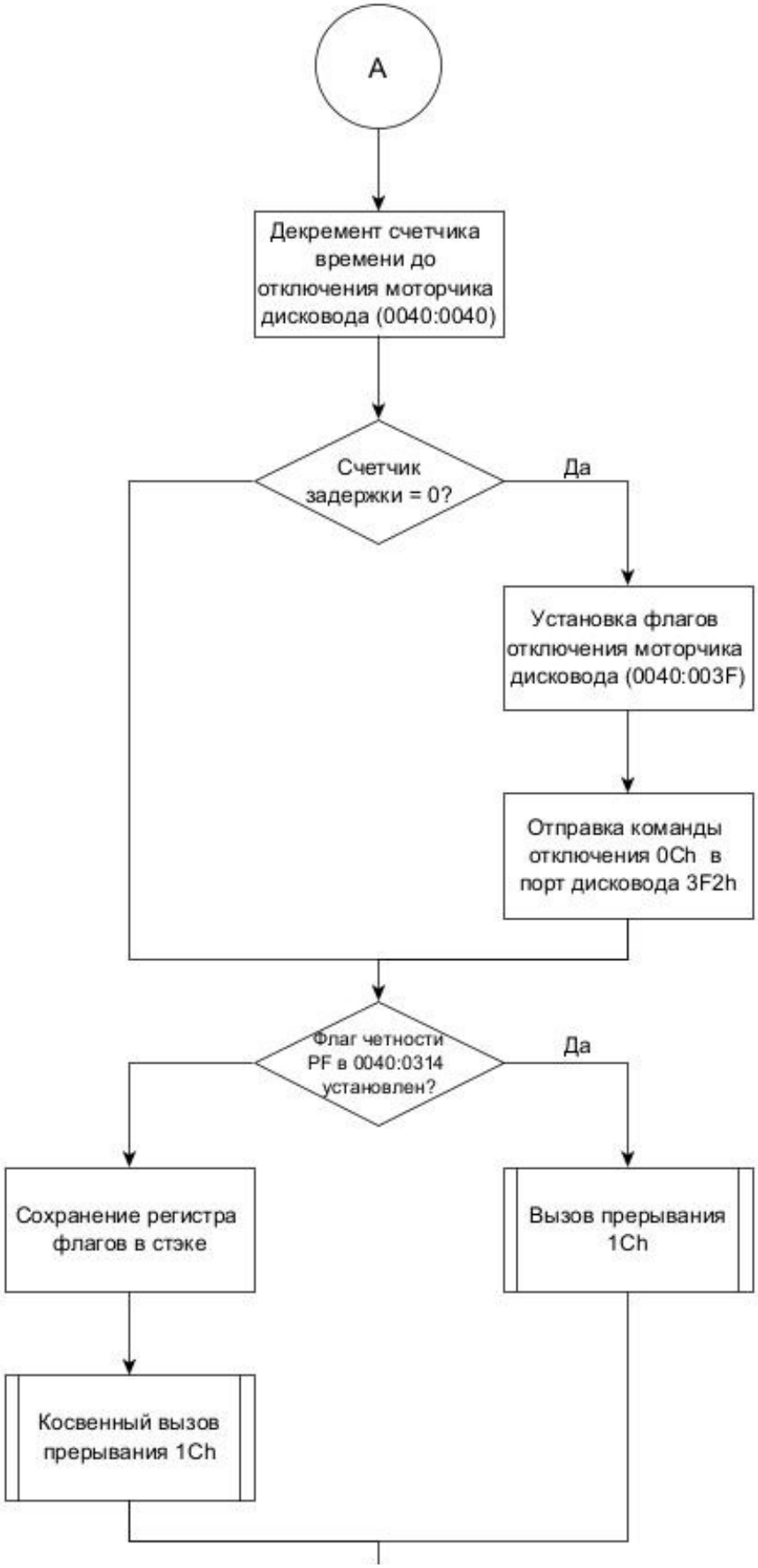
```
36      ; Декремент времени, оставшегося до выключения моторчика дисковода
37      020A:077D  FE 0E 0040               dec byte ptr ds:[40h]    ;
            (0040:0040=81h)
38      020A:0781  75 0B                 jnz loc_3           ; Jump if not zero
39      ; Логический И для содержимого по адресу 0000:043F и F0 - установка флаг
            ов для отключения двигателя
40      020A:0783  80 26 003F F0            and byte ptr ds:[3Fh],0F0h  ;
            (0040:003F=0)
41      ; al = 0Ch, dx = 03F2h
42      ; dx - номер порта дисковода - 3F2h, отправляется команда 00001100 для о
            тключения моторчика
43      020A:0788  B0 0C                 mov al,0Ch ; al = 1100
44      020A:078A  BA 03F2               mov dx,3F2h
45      020A:078D  EE                    out dx,al           ; port 3F2h, dsk0
            contrl output
46
47  020A:078E           loc_3:
48      020A:078E  58                    pop ax
49      ; Проверка, установлен ли флаг PF
50      020A:078F  F7 06 0314 0004          test    word ptr ds:[314h],4    ;
            (0040:0314=3200h)
51      020A:0795  75 0C                 jnz loc_4           ; Вызов прерывания 1
            Ch
52      ; Сохранение состояния регистра флагов
53      020A:0797  9F                    lahf                ; Load ah from flags
54      020A:0798  86 E0                 xchg    ah,al
55      020A:079A  50                    push    ax
56      ; Косвенный вызов прерывания 1Ch (70h / 4 = 1Ch)
57      020A:079B  26: FF 1E 0070           call    dword ptr es:[70h]  ;
            (0000:0070=6ADh)
58      020A:07A0  EB 03                 jmp short loc_5     ; (07A5)
59      020A:07A2  90                    nop
60
61  020A:07A3           loc_4:
62  020A:07A3  CD 1C                int 1Ch          ; Timer break (call each 18
        .2ms)
63
64  020A:07A5           loc_5:
65      ; Остановить прерывания
66      020A:07A5  E8 0011               call    sub_1           ; (07B9)
67      ; Сбросить контроллер прерываний
68      020A:07A8  B0 20                 mov al,20h          ; ' '
69      020A:07AA  E6 20                 out 20h,al          ; port 20h,     al =
            20h, end of interrupt
70      ; Восстановить аппаратный контекст (dx, ax, ds, es)
71      020A:07AC  5A                    pop dx
```

```
72      020A:07AD  58                  pop ax
73      020A:07AE  1F                  pop ds
74      020A:07AF  07                  pop es
75      ; Переход на 164h байта назад - 020A:064Ch, завершение прерывания
76      020A:07B0  E9 FE99             jmp $-164h
77
78          020A:064C           loc_1:
79              ; Сохранить ds, ax
80              020A:064C  1E                   push    ds
81              020A:064D  50                   push    ax
82              ; ds = 40h
83              020A:064E  B8 0040              mov ax,40h
84              020A:0651  8E D8                mov ds,ax
85
86              020A:0653  F7 06 0314 2400          test    word ptr ds:[314h
                    ],2400h     ; (0040:0314=3200h)
87              020A:0659  75 4F                jnz loc_9           ; Jump if
                    not zero
88              ; .....
89              020A:06AA           loc_9:
90              ; Загрузить ax, ds
91              020A:06AA  58                   pop ax
92              020A:06AB  1F                   pop ds
93              ; Завершить прерывание
94              020A:06AC  CF                   iret                ; Interrupt
                    return
```

Схема алгоритма обработчика INT 8h




Младшая часть счетчика = 0?
Да
Инкремент старшей части счётчика реального времени (DS:[6Eh])
Старшая часть счетчика = 18h?
Младшая часть счетчика = 0B0h?
Да
Обнуление счетчика таймера
Установка флага прошедших суток (0040:0070)
AL = 8 (Тогда AX = 8)
A

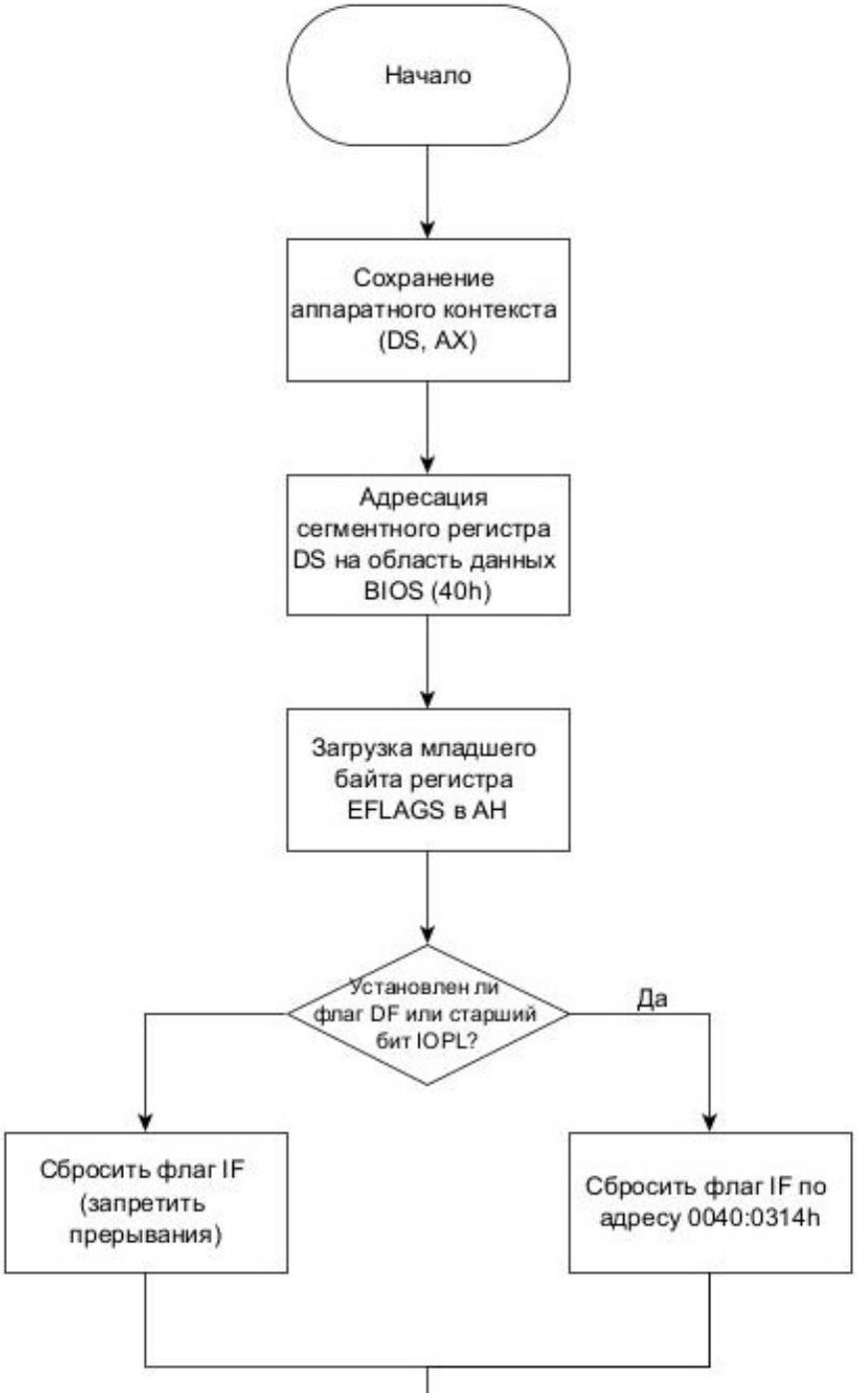

A
Декремент счетчика времени до отключения моторчика дисковода (0040:0040)
Счетчик задержки = 0?
Да
Установка флагов отключения моторчика дисковода (0040:003F)
Отправка команды отключения 0Ch в порт дисковода 3F2h
Флаг четности PF в 0040:0314 установлен?
Да
Сохранение регистра флагов в стэке
Вызов прерывания 1Ch
Косвенный вызов прерывания 1Ch


Вызов процедуры sub_1
Сброс контроллера прерываний (команда 20h в порт 20h)
Восстановление аппаратного контекста (AX, DX, DS, ES)
Конец

Листинг процедуры subroutine

```
sub_1       proc    near
    ; Сохранение аппаратного контекста
    020A:07B9  1E                   push    ds
    020A:07BA  50                   push    ax

    ; Установка сегментного регистра DS на область данных BIOS
    020A:07BB  B8 0040              mov ax,40h ; ax = 40h
    020A:07BE  8E D8                mov ds,ax ; ds = 40h

    ; Загрузка регистра флагов в регистр ah
    020A:07C0  9F                   lahf

    ; Проверка, подняты ли флаг PF или старший бит IOPL
    020A:07C1  F7 06 0314 2400          test    word ptr ds:[314h],2400h
       ; (0040:0314=3200h)

    ; Отключить прерывания, если флаги подняты
    020A:07C7  75 0C                jnz loc_7           ; Jump if not zero -
       проверяет флаги ZF - если результат предыдущей операции = 0

    ; Шина данных блокируется на время выполнения следующей команды
    ; Обращение к памяти происходит дважды - Чтение и запись по адресу
       0040:0314
    020A:07C9  F0> 81 26 0314 FDFF      lock and word ptr ds:[314h],0FDFFh
       ; (0040:0314=3200h)

    020A:07D0           loc_6:
    ; Установка регистра флагов
    020A:07D0  9E                   sahf                ; Store ah into
       flags
    ; Восстановление аппаратного контекста
    020A:07D1  58                   pop ax
    020A:07D2  1F                   pop ds
    ; Завершение процедуры
    020A:07D3  EB 03                jmp short loc_8     ; (07D8)

    ; loc_7 - Сброс Interrupt Enable Flag - отключает прерывания
    020A:07D5           loc_7:
    020A:07D5  FA                   cli             ; Disable interrupts
    020A:07D6  EB F8                jmp short loc_6     ; (07D0)
    020A:07D8           loc_8:
    020A:07D8  C3                   retn
sub_1       endp
```

Схема алгоритма процедуры subroutine




Загрузка AH в младший байт регистра EFLAGS
Восстановление аппаратного контекста (DS, AX)
Конец

Функции прерывания int 8h

- Инкремент счётчика реального времени по известному адресу в области данных BIOS
- Вызов пользовательского прерывания 1Ch.
- Декремент счётчика времени до отключения моторчика дисковода. Посылка команды в порт на отключение дисковода по истечении двух секунд.

Вывод

В ходе работы были вычислены адреса в памяти и дизассемблированы коды обработчика прерывания int 8h и подпрограммы subroutine, которая вызывается из кода обработчика. Были построены схемы алгоритмов обработчика прерывания int8h и подпрограммы subroutine.