**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"
# "Дизассемблирование INT 8h"

**Студент:** Шахнович Д.С.

**Группа:** ИУ7-52Б

**Преподаватель:** Рязанова Н.Ю.

Москва, 2024 г.

# 1 Листинги дизассемблированного кода

## 1.1 Листинг int 8h

```
      Temp.lst                Sourcer v5.10   12-Sep-24   2:18 pm   Page 1

; Вызов сабрутины
020A:0746  E8 0070              call    sub_1           ; (07B9)

; Сохранение регистров
020A:0749  06                   push    es
020A:074A  1E                   push    ds
020A:074B  50                   push    ax
020A:074C  52                   push    dx

; Установка ds - сегмент bios, es - сегмент векторов прерывания
020A:074D  B8 0040              mov ax,40h
020A:0750  8E D8                mov ds,ax
020A:0752  33 C0                xor ax,ax           ; Zero register
020A:0754  8E C0                mov es,ax

; Инкремент младшего слова счётчика тиков
020A:0756  FF 06 006C               inc word ptr ds:[6Ch]   ; (0040:006C=4
   DD4h)
020A:075A  75 04                jnz loc_1           ; Jump if not zero

; Если младшее слово обнулилось, то инкремент старшего слова счётчика тиков
020A:075C  FF 06 006E               inc word ptr ds:[6Eh]   ; (0040:006E=0Eh
   )
020A:0760           loc_1:

Если старшее слово равно 18h(24) и младшее равно B0h(176) то это значит, что
    прошёл день
020A:0760  83 3E 006E 18            cmp word ptr ds:[6Eh],18h   ; (0040:006E
   =0Eh)
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
   =4DD4h)
020A:076D  75 0D                jne loc_2           ; Jump if not equal
; Зануляем младшее и старшее слова счётчика тиков
020A:076F  A3 006E              mov word ptr ds:[6Eh],ax    ; (0040:006E=0Eh
   )
020A:0772  A3 006C              mov word ptr ds:[6Ch],ax    ; (0040:006C=4
   DD4h)

; Записываем значение 1 по 0040h:0070h
020A:0775  C6 06 0070 01            mov byte ptr ds:[70h],1 ; (0040:0070=0)

; Устанавливаем 3-й бит в al
020A:077A  0C 08                or  al,8
020A:077C           loc_2:

; Работа с моторчиком дисковода
020A:077C  50                   push    ax
; Декремент счётчика времени до отключения моторчика
020A:077D  FE 0E 0040               dec byte ptr ds:[40h]   ; (0040:0040=0
   E2h)
; Если счётчик обнулился, то
020A:0781  75 0B                jnz loc_3           ; Jump if not zero
; Сбрасываем флаг работы моторчика
```

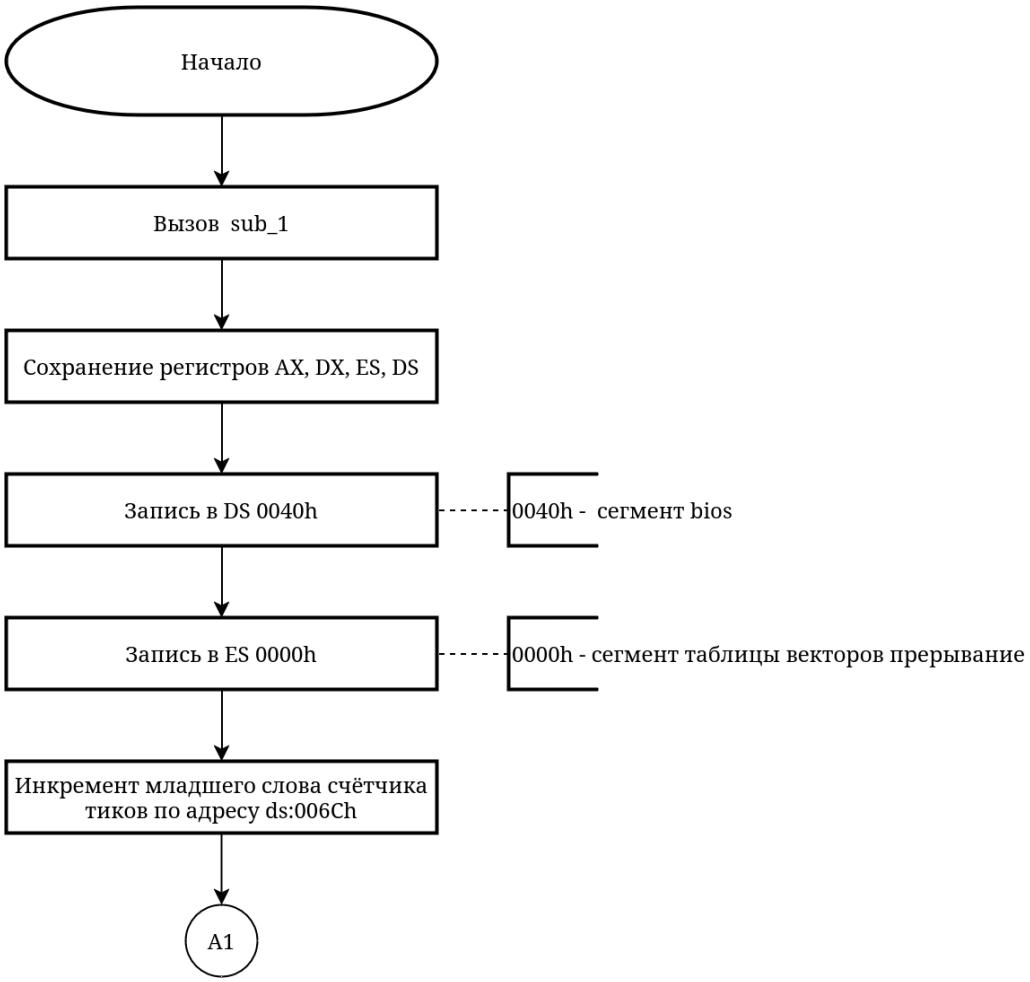
```
020A:0783  80 26 003F F0            and byte ptr ds:[3Fh],0F0h  ; (0040:003F=0)
; Отправляем команду отключения моторчика дисквода 0Ch на порт 3F2h
020A:0788  B0 0C                    mov al,0Ch
020A:078A  BA 03F2                  mov dx,3F2h
020A:078D  EE                       out dx,al           ; port 3F2h, dsk0 contrl output
020A:078E            loc_3:
020A:078E  58                       pop ax

; Проверка флага чётности(PF - 2-й бит)
020A:078F  F7 06 0314 0004          test    word ptr ds:[314h],4    ; (0040:0314=3200h)
020A:0795  75 0C                    jnz loc_4           ; Jump if not zero
; Загрузка младшего байта регистра флагов a ah
020A:0797  9F                       lahf                ; Load ah from flags
; Смена ah, al
020A:0798  86 E0                    xchg    ah,al
; Сохраняем регистр флагов
020A:079A  50                       push    ax
; Косвенный вызов прерывания 1Ch
020A:079B  26: FF 1E 0070           call    dword ptr es:[70h]  ; (0000:0070=6ADh)
020A:07A0  EB 03                    jmp short loc_5     ; (07A5)
020A:07A2  90                       nop
020A:07A3            loc_4:
; Прямой вызов 1Ch, если прерывания не запрещены
020A:07A3  CD 1C                    int 1Ch         ; Timer break (call each 18.2ms)
020A:07A5            loc_5:
; Вызов сабрутины
020A:07A5  E8 0011                  call    sub_1           ; (07B9)
; Завершение прерывания(сброс контроллера прерываний), разрешения прерывания с более низкими приоритетами
020A:07A8  B0 20                    mov al,20h          ; ' '
020A:07AA  E6 20                    out 20h,al          ; port 20h, 8259-1 int command
;  al = 20h, end of interrupt
; Восстановление значений регистров
020A:07AC  5A                       pop dx
020A:07AD  58                       pop ax
020A:07AE  1F                       pop ds
020A:07AF  07                       pop es

020A:07B0  E9 FE99                  jmp $-164h  ; 07B0h - 164h = 064Ch
; ...
020A:064C  1E                       push    ds
020A:064D  50                       push    ax
; ...
020A:06AA  58                       pop ax
020A:06AB  1F                       pop ds
020A:06AC  CF                       iret ; возврат из прерывания
```

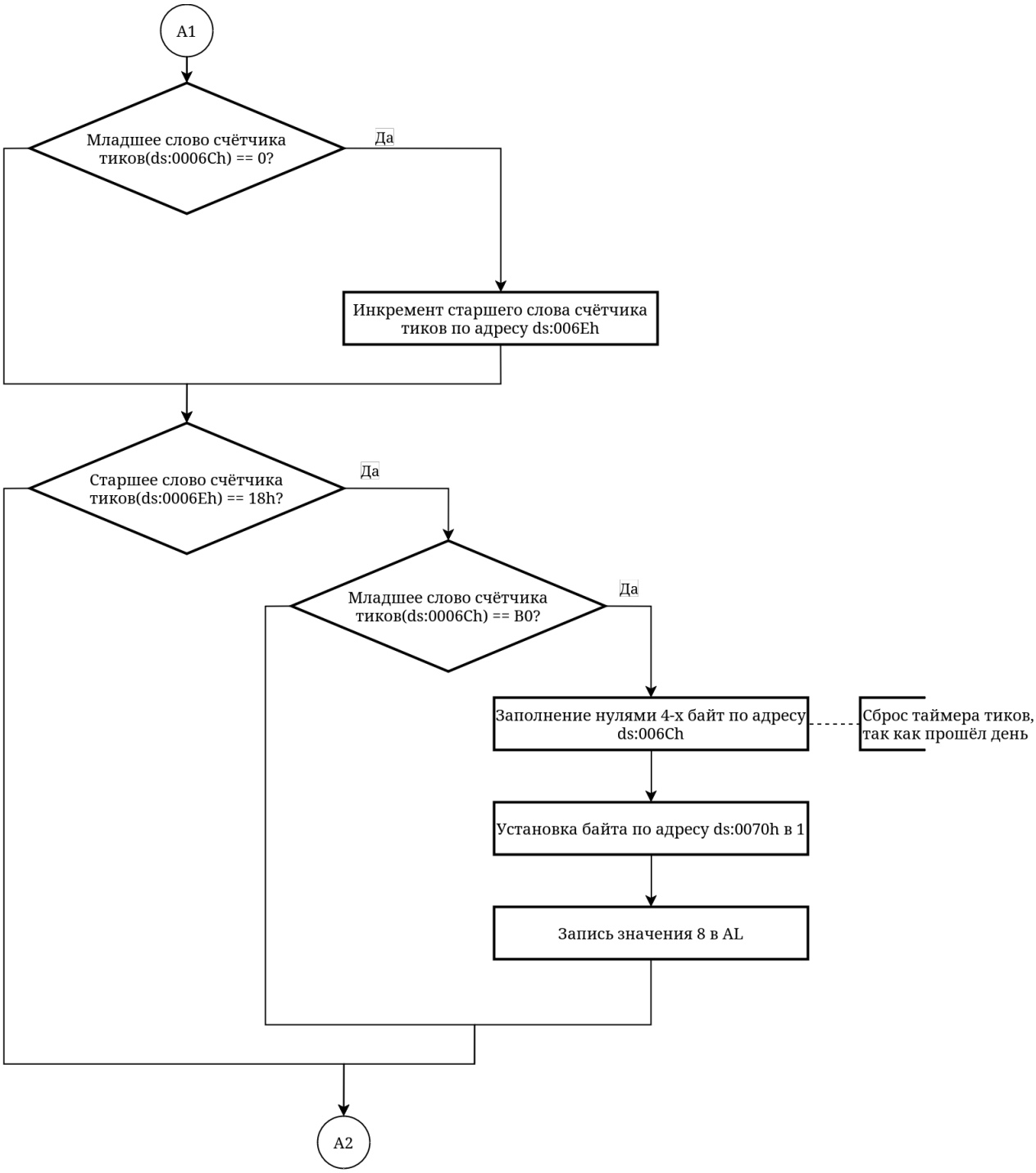
## 1.2 Листинг процедуры sub_1

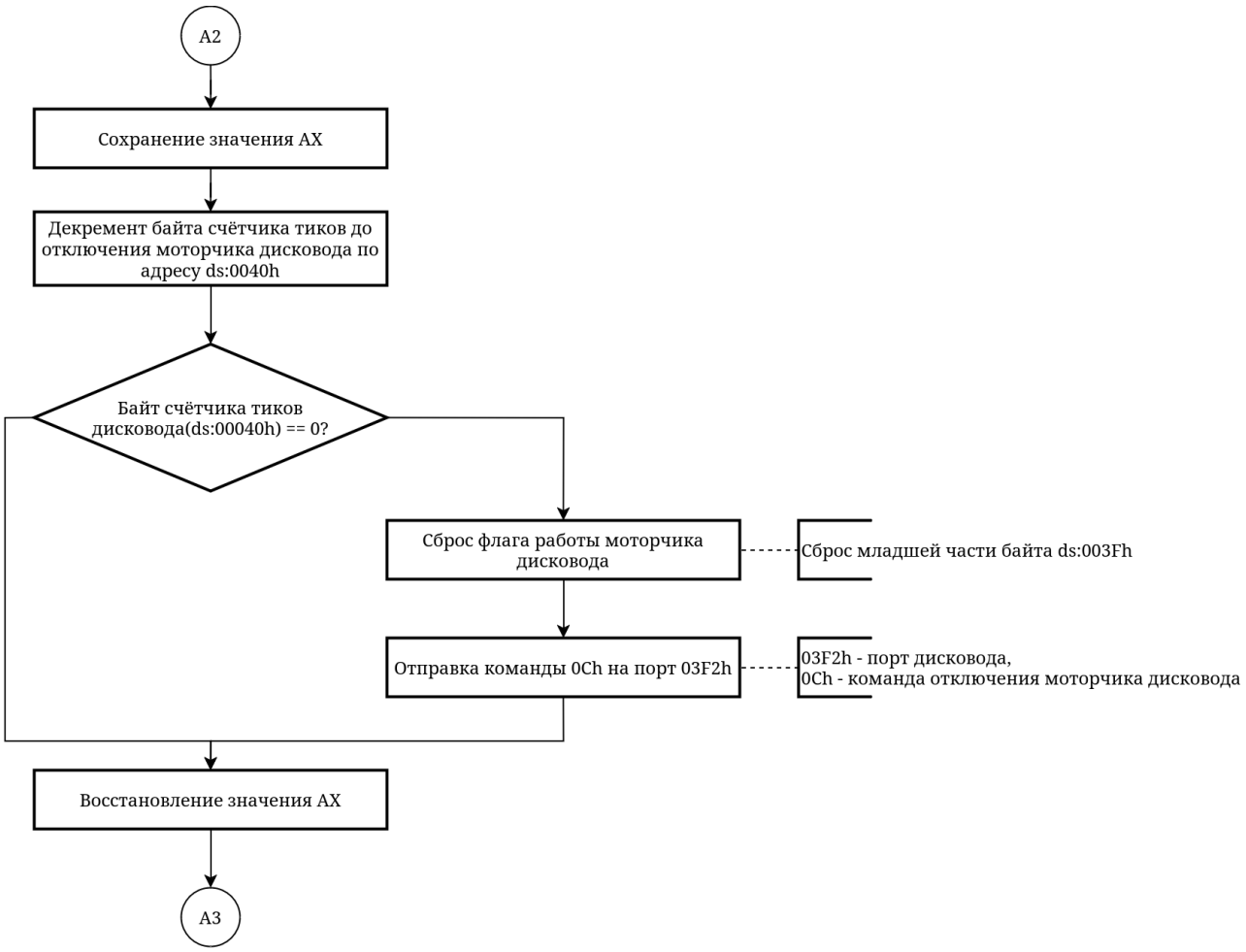
```
                sub_1       proc    near
; Сохранение значений регистров
020A:07B9  1E                   push    ds
020A:07BA  50                   push    ax
; Установка ds - сегмент bios
020A:07BB  B8 0040              mov ax,40h
020A:07BE  8E D8                mov ds,ax

; Загрузка в ah младшего бита регистра флагов
020A:07C0  9F                   lahf                ; Load ah from flags
; Проверка DF и старшего бита IOPL
020A:07C1  F7 06 0314 2400          test    word ptr ds:[314h],2400h    ;
    (0040:0314=3200h)
020A:07C7  75 0C                jnz loc_7           ; Jump if not zero
; Сброс флага прерываний IF в BIOS
020A:07C9  F0> 81 26 0314 FDFF                              lock and word ptr
    ds:[314h],0FDFFh    ; (0040:0314=3200h)
020A:07D0           loc_6:
; Загрузка из ah младшего бит флагов
020A:07D0  9E                   sahf                ; Store ah into flags
; Восстановление регистров
020A:07D1  58                   pop ax
020A:07D2  1F                   pop ds
020A:07D3  EB 03                jmp short loc_8     ; (07D8)
020A:07D5           loc_7:
; Сброс флага прерываний IF
020A:07D5  FA                   cli             ; Disable interrupts
020A:07D6  EB F8                jmp short loc_6     ; (07D0)
020A:07D8           loc_8:
; Возврат из сабрутины
020A:07D8  C3                   retn
sub_1       endp
```

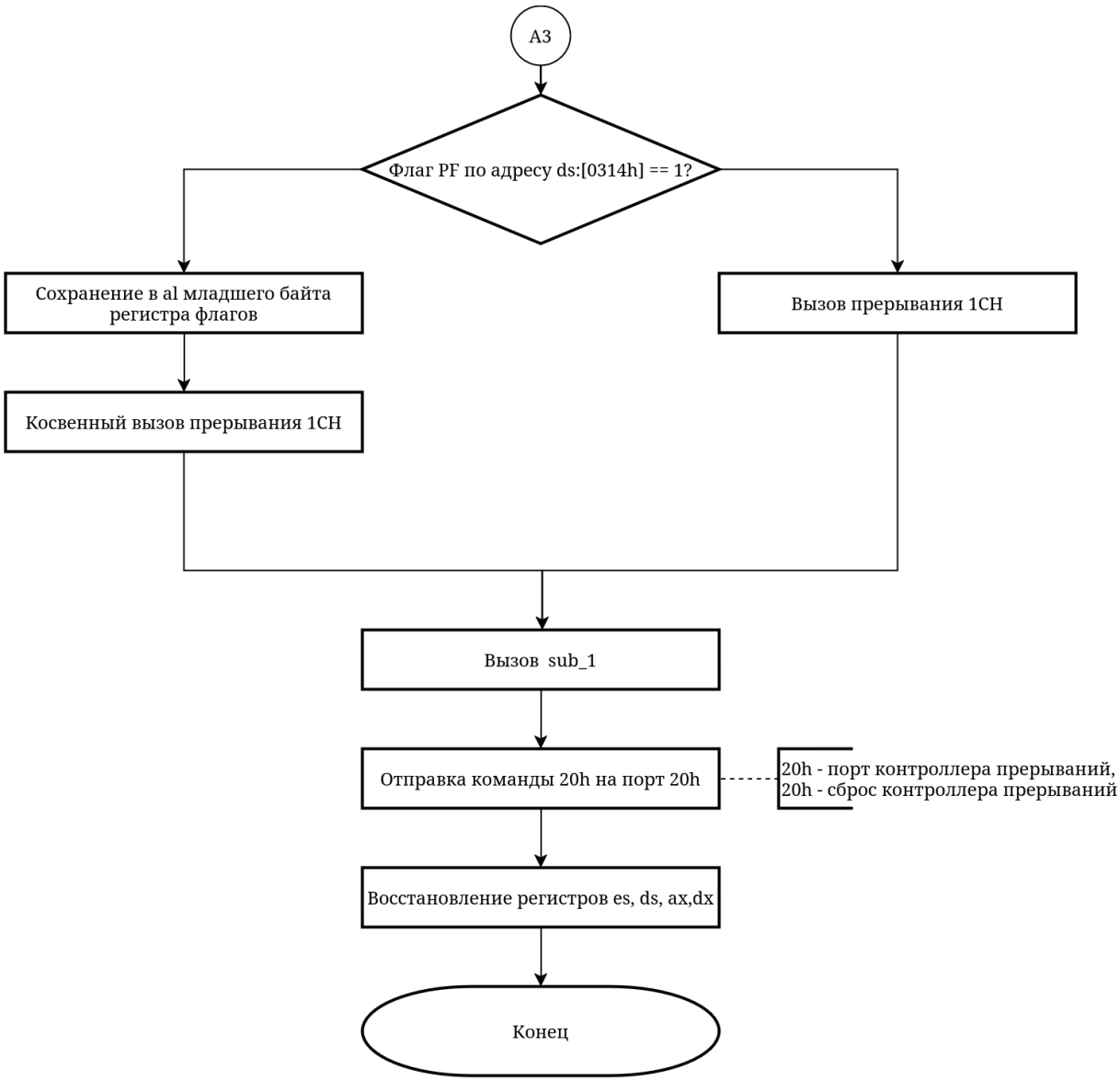
# 2 Схемы алгоритмов

## 2.1 Схема алгоритма int 8h

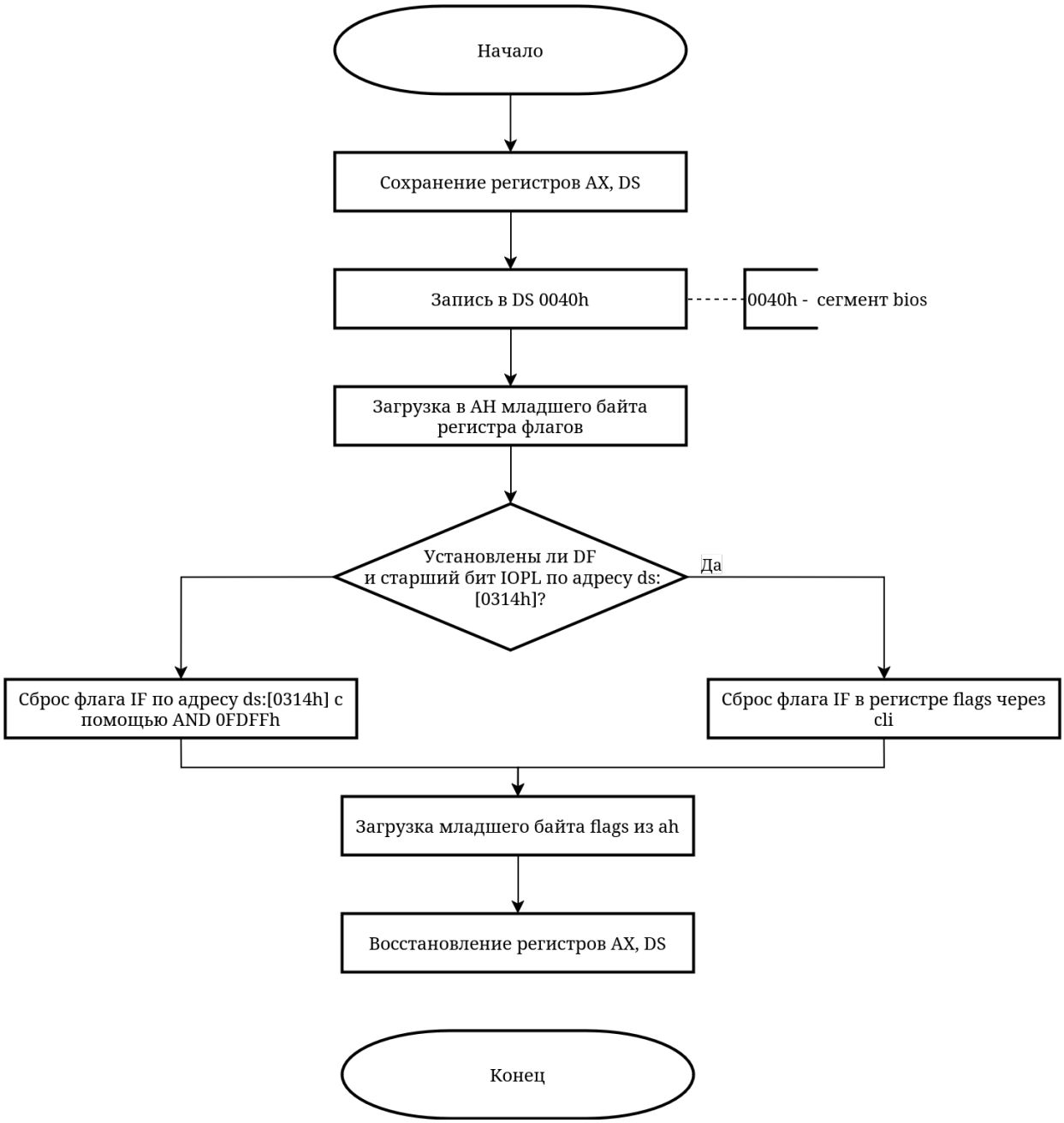



A1
Младшее слово счётчика тиков(ds:0006Ch) == 0?
Да
Инкремент старшего слова счётчика тиков по адресу ds:006Eh
Старшее слово счётчика тиков(ds:0006Eh) == 18h?
Да
Младшее слово счётчика тиков(ds:0006Ch) == B0?
Да
Заполнение нулями 4-х байт по адресу ds:006Ch
Сброс таймера тиков, так как прошёл день
Установка байта по адресу ds:0070h в 1
Запись значения 8 в AL
A2


A2
Сохранение значения AX
Декремент байта счётчика тиков до отключения моторчика дисковода по адресу ds:0040h
Байт счётчика тиков дисковода(ds:00040h) == 0?
Сброс флага работы моторчика дисковода
Сброс младшей части байта ds:003Fh
Отправка команды 0Ch на порт 03F2h
03F2h - порт дисковода,
0Ch - команда отключения моторчика дисковода
Восстановление значения AX
A3


A3
Флаг PF по адресу ds:[0314h] == 1?
Сохранение в al младшего байта регистра флагов
Косвенный вызов прерывания 1CH
Вызов прерывания 1CH
Вызов sub_1
Отправка команды 20h на порт 20h
20h - порт контроллера прерываний,
20h - сброс контроллера прерываний
Восстановление регистров es, ds, ax,dx
Конец

## 2.2 Схема алгоритма sub_1



# 3 Заключение

В ходе работы мною был разобран и изучен дизассемблированный код обработчика int 8h, выявлены его основные функции:

1. Инкремент счётчика тиков
2. Контроль счётчика тиков при наступлении нового дня
3. Декремент счётчика тиков до выключения моторчика дисковода
4. Выключение моторчика, в случае зануления таймера
5. Вызов обработчика прерывания int 1Ch