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»

НАПРАВЛЕНИЕ ПОДГОТОВКИ «09.03.04 Программная инженерия»

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №1(ЧАСТЬ 1)

Название: Прерывание INT 8h

Дисциплина: Операционные системы

Студент	ИУ7-56Б		Т. А. Казаева
	Группа	Подпись, дата	И. О. Фамилия
Преподаватель			Н. Ю. Рязанова
		Подпись, дата	И. О. Фамилия

Москва, 2021 г.

1. Цель лабораторной работы

Знакомство со средством дизассемблирования – sourcer и с получением дизассемблерного кода ядра операционной системы Windows на примере обработчика прерывания Int 8h в virtual mode – специальном режиме защищенного режима, который эмулирует реальный режим работы вычислительной системы на базе процессоров Intel.

2. Задание

Используя Sourser получить дизассемблерный код обработчика аппаратного прерывания от системного таймера INT 8h. На основе полученного кода составить алгоритм работы обработчика INT 8h.

3. Листинги кода

```
020A:0746  E8 0070              call    sub_9                   ; (07B9)
; Сохранение значений регистров ES, DS, AX, DX
020A:0749  06                   push    es
020A:074A  1E                   push    ds
020A:074B  50                   push    ax
020A:074C  52                   push    dx
;;   Загрузка в DS (через буфер AX) адреса области данных BIOS
020A:074D  B8 0040          mov ax,40h
020A:0750  8E D8                mov ds,ax
020A:0752  33 C0                xor ax,ax                       ; Zero
   register
; Установка адреса начала таблицы векторов прерывания в ES
020A:0754  8E C0                mov es,ax
;; Инкремент счетчика таймера (располагается по адресу  0040:006Ch), прошл
   а секунда
020A:0756  FF 06 006C           inc word ptr ds:[6Ch]           ; (0040:006C
   =0CFC1h)
```

```
020A:075A  75 04                jnz loc_19                      ; Jump if not zero
;; Инкремент двух старших байтов счетчика таймера(располагается по адресу 0040:006Eh), прошел час
020A:075C  FF 06 006E           inc word ptr ds:[6Eh]           ; (0040:006E=16h)
020A:0760           loc_19:
;; Проверка на то, прошли ли сутки: 18h = 24
020A:0760  83 3E 006E 18        cmp word ptr ds:[6Eh],18h       ; (0040:006E=16h)
020A:0765  75 15                jne loc_20                      ; Jump if not equal
;;
020A:0767  81 3E 006C 00B0      cmp word ptr ds:[6Ch],0B0h      ; (0040:006C=0CFC1h)
020A:076D  75 0D                jne loc_20                      ; Jump if not equal
;; Сутки прошли - зануление счетчика таймера
020A:076F  A3 006E              mov word ptr ds:[6Eh],ax        ; (0040:006E=16h)
020A:0772  A3 006C              mov word ptr ds:[6Ch],ax        ; (0040:006C=0CFC1h)
020A:0775  C6 06 0070 01        mov byte ptr ds:[70h],1         ; (0040:0070=0)
;; Разрешение прямого доступа к памяти и прерываний
020A:077A  0C 08                or  al,8
020A:077C           loc_20:
020A:077C  50                   push    ax
;; Декремент счетчика отключения моторчика дисковода
020A:077D  FE 0E 0040           dec byte ptr ds:[40h]           ; (0040:0040=60h)
020A:0781  75 0B                jnz loc_21                  ; Jump if not zero
;; Установка флагов, отвечающих за отключение моторчика дисковода
020A:0783  80 26 003F F0            and byte ptr ds:[3Fh],0F0h  ; (0040:003F=0)
;; Посылка команды 0Ch в порт 3F2h для отключения моторчика дисковвода
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_21:
020A:078E  58                   pop ax
;; Проверка на четность - второй бит счетчика отвечает за то, будет ли выключен моторчик,
;; флаги в этом случае изменяться не должны и вызывать прерывание нужно косвенно (при вызове
;; int происходит пуш флагов в стек), иначе прерывание вызывается напрямую
```

```
020A:078F  F7 06 0314 0004          test    word ptr ds:[314h],4    ;
    (0040:0314=3200h)
020A:0795  75 0C                    jnz loc_22                  ; Jump if not
    zero
020A:0797  9F                       lahf                        ; Load ah from
    flags
;; Теперь AX = 08, а AH - младший байт регистра флагов
020A:0798  86 E0                    xchg    ah,al
020A:079A  50                       push    ax
;; Вызов прерывания 1Ch с помощью его адреса в таблице прерываний
020A:079B  26: FF 1E 0070           call    dword ptr es:[70h]  ;
    (0000:0070=6ADh)
020A:07A0  EB 03                    jmp short loc_23        ; (07A5)
020A:07A2  90                       nop
020A:07A3           loc_22:
020A:07A3  CD 1C                    int 1Ch         ; Timer break (call each
    18.2ms)
020A:07A5           loc_23:
020A:07A5  E8 0011                  call    sub_9           ; (07B9)
;; Сброс контроллера прерываний, иначе менее приоритетные прерывания будут
    игнорироваться
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
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 loc_3           ; (064C)

020A:064C           loc_3:
020A:064C  1E                       push    ds
020A:064D  50                       push    ax
;; <...>
020A:06AA  58                       pop ax
020A:06AB  1F                       pop ds
020A:06AC  CF                       iret                ; Interrupt return
```

Листинг 1 – Листинг прерывания INT 8h

```
                sub_9       proc    near
020A:07B9  1E                       push    ds
020A:07BA  50                       push    ax
020A:07BB  B8 0040                  mov ax,40h
020A:07BE  8E D8                    mov ds,ax
020A:07C0  9F                       lahf
           ; Load ah from flags
```

```
;; Проверка флага DF или старшего бита IPOL
;; если хоть один занулен, сбрасывается флаг прерываний (cil)
020A:07C1  F7 06 0314 2400      test    word ptr ds:[314h],2400h
           ; (0040:0314=3200h)
020A:07C7  75 0C                jnz loc_4
           ; Jump if not zero
;; Сброс IF
;; lock - перекрытие шины данных, чтобы процесс не использовал память во вр
   емя выполнения команды
020A:07C9  F0> 81 26 0314 FDFF  lock    and word ptr ds:[314h],0FDFFh
           ; (0040:0314=3200h)
020A:07D0           loc_3:
020A:07D0  9E                   sahf
           ; Store ah into flags
020A:07D1  58                   pop ax
020A:07D2  1F                   pop ds
020A:07D3  EB 03                jmp short loc_5
           ; (07D8)
020A:07D5           loc_4:
;; cli - сброс IF
020A:07D5  FA                   cli
           ; Disable interrupts
020A:07D6  EB F8                jmp short loc_3
           ; (07D0)
020A:07D8           loc_5:
020A:07D8  C3                   retn
sub_9                           endp
```

Листинг 2 – Листинг процедуры sub_9

4. Схемы алгоритмов

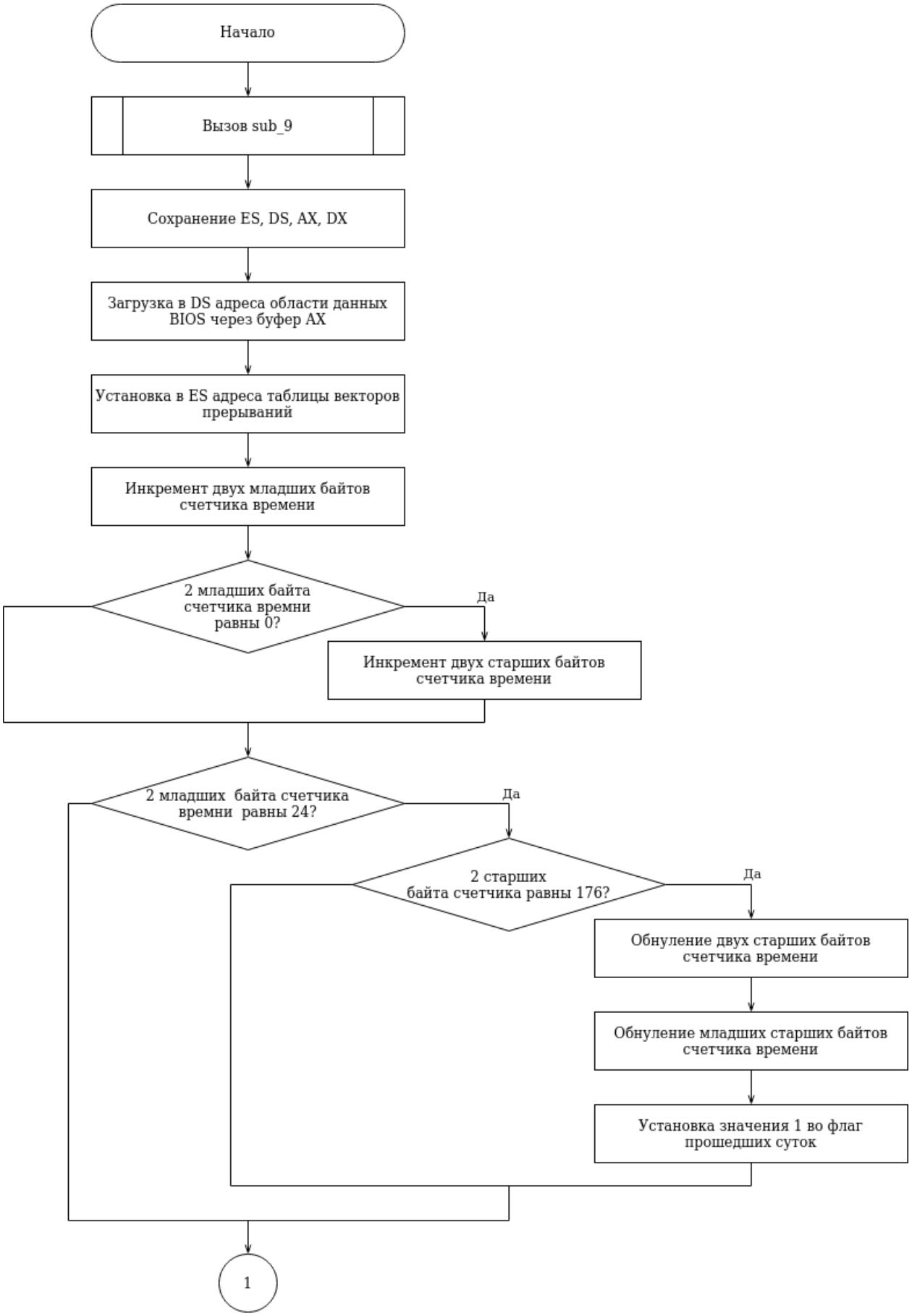

Рисунок 1 – Схема алгоритма прерывания INT 8h

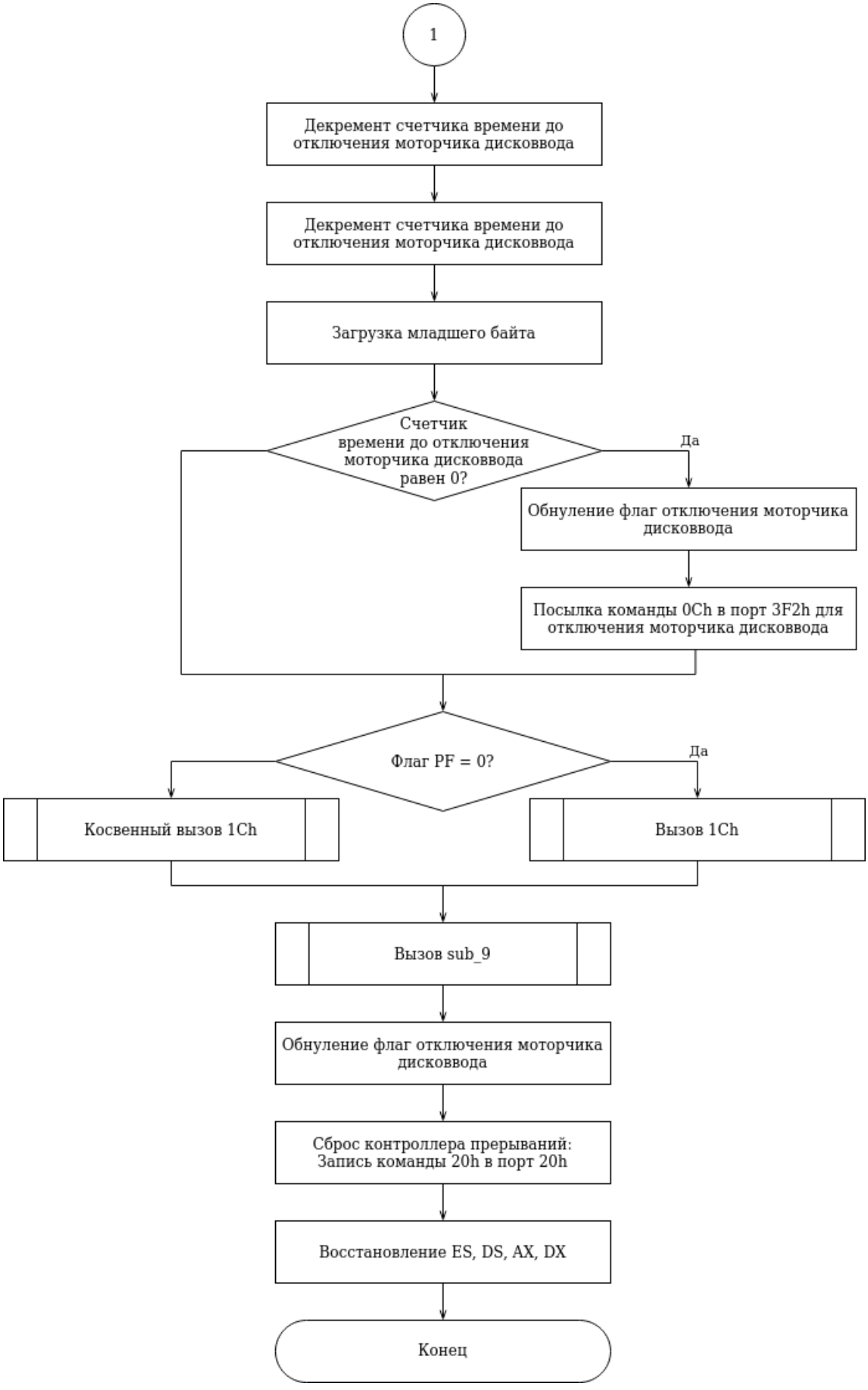

Рисунок 2 – Схема алгоритма прерывания INT 8h

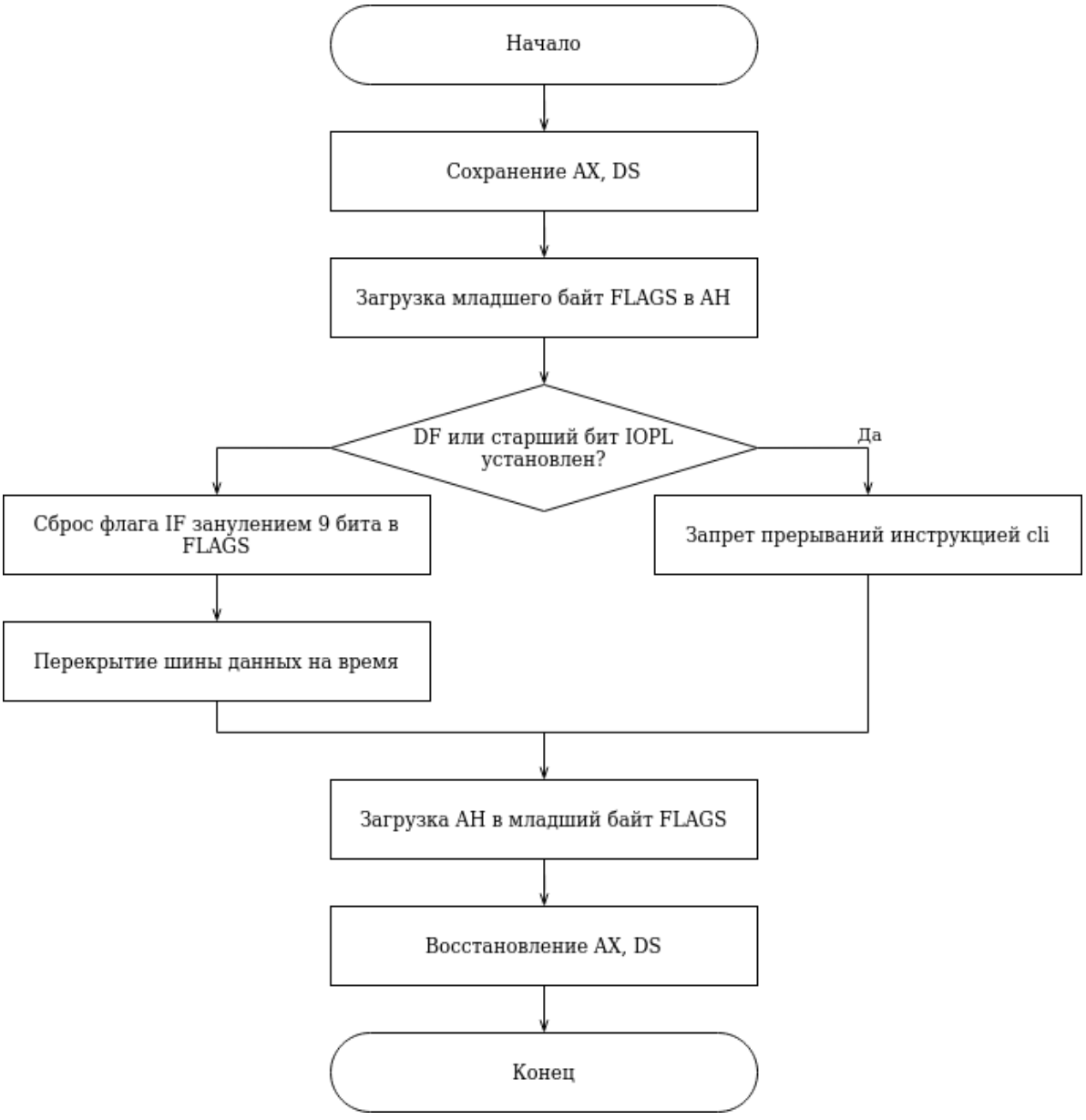

Рисунок 3 – Схема процедуры sub_9