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»

Отчёт

по лабораторной работе №1

Название: Дизассемблирование прерывания INT 8h

Дисциплина: Операционные системы

Студент ИУ7-55Б (Группа) _____ (Подпись, дата) А.К. Клименко (И.О. Фамилия)

Преподователь _____ (Подпись, дата) Н.Ю. Рязанова (И.О. Фамилия)

Москва, 2021

1 Код обработчика прерываний от системного таймера

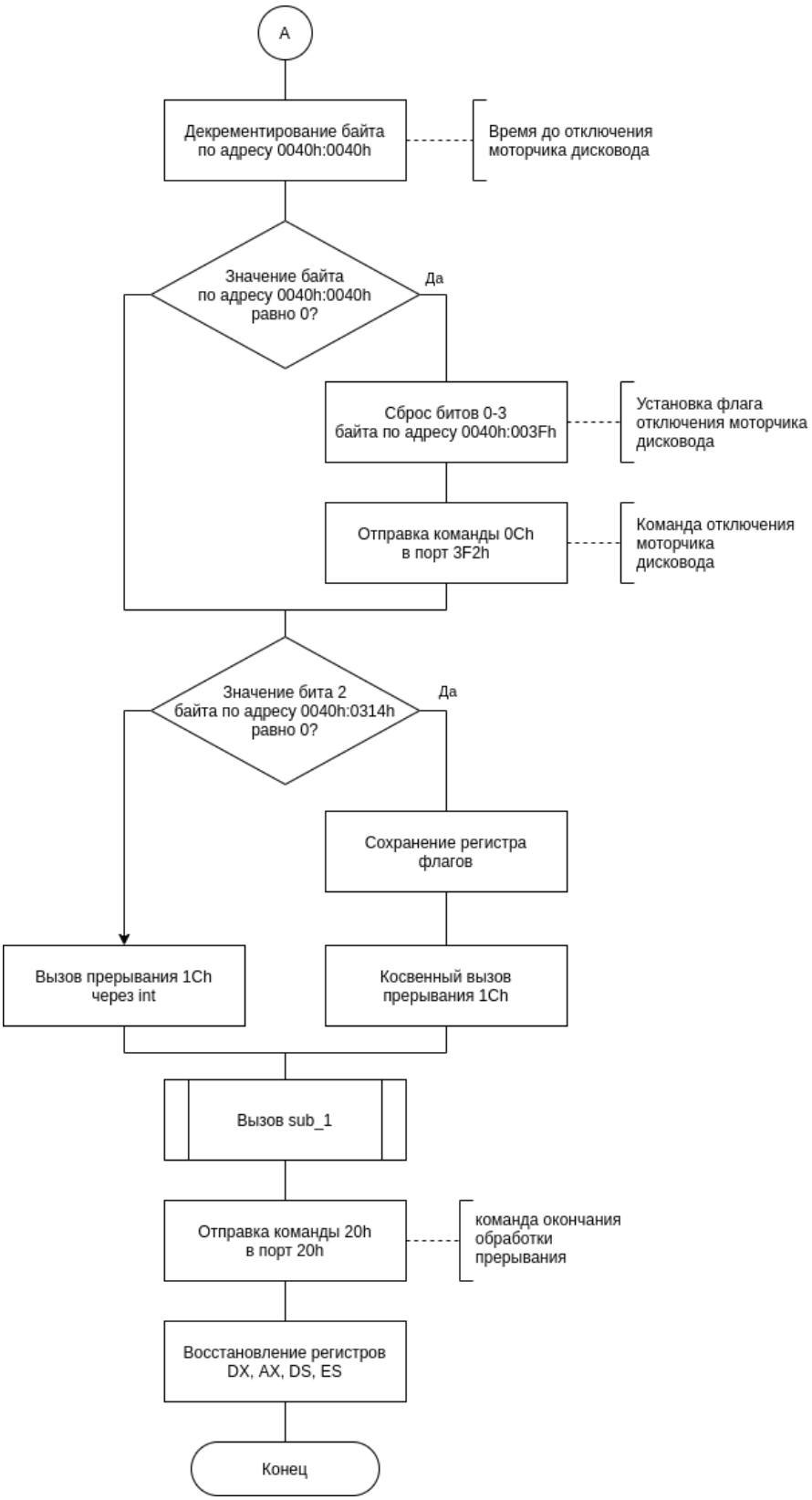
```
1         Temp.lst  Sourcer Listing v2.13    11-Sep-21   5:28 am
2
3     ;================;
4     ; INTERUPTION 8h ;
5     ;================;
6  020E:0746  E8 0070              call    sub_1   ; (07B9)
7  020E:0749  06                   push    es
8  020E:074A  1E                   push    ds
9  020E:074B  50                   push    ax
10 020E:074C  52                   push    dx
11 020E:074D  B8 0040              mov ax,40h
12 020E:0750  8E D8                mov ds,ax
13 020E:0752  33 C0                xor ax,ax       ; Zero register
14 020E:0754  8E C0                mov es,ax
15 020E:0756  FF 06 006C           inc word ptr ds:[6Ch]
16 020E:075A  75 04                jnz loc_16
17 020E:075C  FF 06 006E           inc word ptr ds:[6Eh]
18             loc_16:                         ;  xref 020E:075A
19 020E:0760  83 3E 006E 18        cmp word ptr ds:[6Eh],18h
20 020E:0765  75 15                jne loc_17
21 020E:0767  81 3E 006C 00B0      cmp word ptr ds:[6Ch],0B0h
22 020E:076D  75 0D                jne loc_17
23 020E:076F  A3 006E              mov word ptr ds:[6Eh],ax
24 020E:0772  A3 006C              mov word ptr ds:[6Ch],ax
25 020E:0775  C6 06 0070 01        mov byte ptr ds:[70h],1
26 020E:077A  0C 08                or  al,8
27             loc_17:                         ;  xref 020E:0765, 076D
28 020E:077C  50                   push    ax
29 020E:077D  FE 0E 0040           dec byte ptr ds:[40h]
30 020E:0781  75 0B                jnz loc_18
31 020E:0783  80 26 003F F0        and byte ptr ds:[3Fh],0F0h
32 020E:0788  B0 0C                mov al,0Ch
33 020E:078A  BA 03F2              mov dx,3F2h
34 020E:078D  EE                   out dx,al       ; port 3F2h, dsk0 contrl output
35             loc_18:                         ;  xref 020E:0781
36 020E:078E  58                   pop ax
37 020E:078F  F7 06 0314 0004      test    word ptr ds:[314h],4
38                                             ; (0040:0314=5200h)
39 020E:0795  75 0C                jnz loc_19      ; Jump if not zero
40 020E:0797  9F                   lahf            ; Load ah from flags
41 020E:0798  86 E0                xchg    ah,al
42 020E:079A  50                   push    ax
43 020E:079B  26: FF 1E 0070       call    dword ptr es:[70h]
44                                             ; (0000:0070=6ADh)
45 020E:07A0  EB 03                jmp short loc_20; (07A5)
```

```
020E:07A2  90                  nop
            loc_19:                                 ;  xref 020E:0795
020E:07A3  CD 1C               int 1Ch              ; Timer break (call each 18.2ms)
            loc_20:                                 ;  xref 020E:07A0
020E:07A5  E8 0011             call    sub_1        ; (07B9)
020E:07A8  B0 20               mov al,20h
020E:07AA  E6 20               out 20h,al           ; port 20h, 8259-1 int command
    ;  al = 20h, end of interrupt
020E:07AC  5A                  pop dx
020E:07AD  58                  pop ax
020E:07AE  1F                  pop ds
020E:07AF  07                  pop es
020E:07B0  E9 FE99             jmp $-164h           ; (06AC)
...
020E:06AC  CF                  iret                 ; Interuption return

    ;==============================;
    ; SUBROUTINE                   ;
    ; Called from: 020E:0746, 07A5 ;
    ;==============================;
                sub_1       proc    near
020E:07B9  1E                  push    ds
020E:07BA  50                  push    ax
020E:07BB  B8 0040             mov ax,40h
020E:07BE  8E D8               mov ds,ax
020E:07C0  9F                  lahf                 ; Load ah from flags
020E:07C1  F7 06 0314 2400     test word ptr ds:[314h],2400h
                                                    ; (0040:0314=5200h)
020E:07C7  75 0C               jnz loc_22           ; Jump if not zero
020E:07C9  F0 81 26 0314 FDFF  lock and word ptr ds:[314h],0FDFFh
                                                    ; (0040:0314=5200h)
            loc_21:                                 ;  xref 020E:07D6
020E:07D0  9E                  sahf                 ; Store ah into flags
020E:07D1  58                  pop ax
020E:07D2  1F                  pop ds
020E:07D3  EB 03               jmp short $+5h
            loc_22:                                 ;  xref 020E:07C7
020E:07D5  FA                  cli                  ; Disable interrupts
020E:07D6  EB F8               jmp short loc_21; (07D0)
                sub_1       endp

020E:07D8  C3                  retn
```

2 Схема алгоритма работы обработчика прерываний


A
Декрементирование байта по адресу 0040h:0040h
Время до отключения моторчика дисковода
Значение байта по адресу 0040h:0040h равно 0?
Да
Сброс битов 0-3 байта по адресу 0040h:003Fh
Установка флага отключения моторчика дисковода
Отправка команды 0Ch в порт 3F2h
Команда отключения моторчика дисковода
Значение бита 2 байта по адресу 0040h:0314h равно 0?
Да
Сохранение регистра флагов
Вызов прерывания 1Ch через int
Косвенный вызов прерывания 1Ch
Вызов sub_1
Отправка команды 20h в порт 20h
команда окончания обработки прерывания
Восстановление регистров DX, AX, DS, ES
Конец

3 Схема алгоритма работы процедуры sub_1