|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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 курсу «Операционные Системы»**

|  |  |
| --- | --- |
| **Тема**  Дизассемблирование INT 8h  **Студент** Якуба Д. В.  **Группа** ИУ7-53Б  **Оценка (баллы) \_\_\_\_\_\_\_\_\_\_\_**  **Преподаватель** Рязанова Н. Ю. |  |

Москва

2020 г.

# Листинг прерывания INT 8h

; Вход в прерывание и вызов подпрограммы sub\_1

020A:0746  E8 0070      ;\*      call    sub\_1           ;\*(07B9)

020A:0746  E8 70 00             db  0E8h, 70h, 00h

; 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

; Инициализация значениями регистров

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 счётчика времени, где 0040:006C - адрес счётчика прерываний от таймера

020A:0756  FF 06 006C               inc word ptr ds:[6Ch]   ; (0040:006C=79C4h)

020A:075A  75 04                jnz loc\_1           ; Jump if not zero

020A:075C  FF 06 006E               inc word ptr ds:[6Eh]   ; (0040:006E=12h)

; Сброс счётчика при наступлении нового дня

; 18h = 24

020A:0760           loc\_1:

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

; B0h = 176

020A:0767  81 3E 006C 00B0          cmp word ptr ds:[6Ch],0B0h  ; (0040:006C=79C4h)

020A:076D  75 0D                jne loc\_2           ; Jump if not equal

; ax = 0, сброс счётчика

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=79C4h)

; Занесение единицы по адресу 0040:0070

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

020A:077A  0C 08                or  al,8

; Сигнал отключения моторчика дисковода

020A:077C           loc\_2:

020A:077C  50                   push    ax

020A:077D  FE 0E 0040               dec byte ptr ds:[40h]   ; (0040:0040=76h)

020A:0781  75 0B                jnz loc\_3           ; Jump if not zero

020A:0783  80 26 003F F0            and byte ptr ds:[3Fh],0F0h  ; (0040:003F=0)

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

; Проверка, можно ли вызвать маскируемые прерывания

020A:078E           loc\_3:

020A:078E  58                   pop ax

020A:078F  F7 06 0314 0004          test    word ptr ds:[314h],4    ; (0040:0314=3300h)

020A:0795  75 0C                jnz loc\_4           ; Jump if not zero

020A:0797  9F                   lahf                ; Load ah from flags

020A:0798  86 E0                xchg    ah,al

020A:079A  50                   push    ax

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

; Вызов пользовательского прерывания по срабатыванию таймера

020A:07A3           loc\_4:

020A:07A3  CD 1C                int 1Ch         ; Timer break (call each 18.2ms)

020A:07A5           loc\_5:

020A:07A5  E8 0011      ;\*      call    sub\_1           ;\*(07B9)

; Сброс контроллера прерываний

020A:07A5  E8 11 00             db  0E8h, 11h, 00h

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 dx, ax, ds, es

020A:07AC  5A                   pop dx

020A:07AD  58                   pop ax

020A:07AE  1F                   pop ds

020A:07AF  07                   pop es

; Переход по метке к завершению работы прерывания

020A:07B0  E9 FE99              jmp $-164h

020A:07B3  C4                   db  0C4h

; Восстановление регистров ax, ds

020A:06AA  58                   pop ax

020A:06AB  1F                   pop ds

; Выход из прерывания

020A:06AC  CF                   iret                ; Interrupt return

# Листинг subroutine

                    sub\_1       proc    near

; Сохранение регистров ds, ax

020A:07B9  1E                   push    ds

020A:07BA  50                   push    ax

020A:07BB  B8 0040              mov ax,40h

020A:07BE  8E D8                mov ds,ax

; Загрузка в ah регистра флагов

020A:07C0  9F                   lahf                ; Load ah from flags

; Проверка, можно ли вызвать макируемые прерывания

020A:07C1  F7 06 0314 2400          test    word ptr ds:[314h],2400h    ; (0040:0314=3300h)

020A:07C7  75 0C                jnz loc\_2           ; Jump if not zero

; Установка флага IF в ноль

020A:07C9  F0> 81 26 0314 FDFF                             lock and word ptr ds:[314h],0FDFFh   ; (0040:0314=3300h)

; Запист содержимого регистра AH в регистр флагов

020A:07D0           loc\_1:

020A:07D0  9E                   sahf                ; Store ah into flags

; Восстановление значений регистров ax, ds

020A:07D1  58                   pop ax

020A:07D2  1F                   pop ds

020A:07D3  EB 03                jmp short loc\_3     ; (07D8)

020A:07D5           loc\_2:

; Выставление запрета на вызов маскируемых прерываний

020A:07D5  FA                   cli             ; Disable interrupts ; Сброс флага IF

020A:07D6  EB F8                jmp short loc\_1     ; (07D0)

; Завершение

020A:07D8           loc\_3:

020A:07D8  C3                   retn

                    sub\_1       endp

# Схема работы обработчика прерывания INT 8h