

УО БГУИР  
Кафедра ЭВМ

Отчет по лабораторной работе  
Тема: " Программирование контроллера прерываний "

Выполнил:  
студент группы 950503  
Полховский А.Ф.

Проверил:  
к.т.н., доцент Одинец Д. Н.

Минск 2021

## 1. Постановка задачи

Написать резидентную программу выполняющую перенос всех векторов аппаратных прерываний ведущего и ведомого контроллера на пользовательские прерывания. При этом необходимо написать обработчики аппаратных прерываний, которые будут установлены на используемые пользовательские прерывания и будут выполнять следующие функции:

1. Выводить на экран в двоичной форме следующие регистры контроллеров прерывания (как ведущего, так и ведомого):

- регистр запросов на прерывания;
- регистр обслуживаемых прерываний;
- регистр масок.

При этом значения регистров должны выводиться всегда в одно и то же место экрана.

Осуществлять переход на стандартные обработчики аппаратных прерываний, для обеспечения нормальной работы компьютера.

## 2. Листинг программы

```
                                org 100h

.model tiny
.stack 100h
.data
msgHello    db  "1 - start resident",10,13,"$"
BLACK       equ 00010000b      ;Атрибут текста - черные буквы на белом фоне
GREEN       equ 00010010b      ;Атрибут текста - красные буквы на белом фоне
.code
mov ah,09h
lea dx,msgHello
int 21h
mov ah,07h
int 21h
cmp al,'1'
je  setInterrupts
IRQ0:                                ;Обработчик IRQ0 - прерывание системного таймера
    int 50h
    int 8h
    iret
IRQ1:                                ;Обработчик IRQ1 - прерывание клавиатуры
    int 50h
    int 09h
    iret
IRQ2:                                ;Обработчик IRQ2 - прерывание для ведомого контроллера
    int 50h
    int 0Ah
    iret
IRQ3:                                ;Обработчик IRQ3 - прерывание COM2
    int 50h
    int 0Bh
    iret
```

IRQ4:	;Обработчик IRQ4 - прерывание COM1
int 50h	
int 0Ch	
iret	
IRQ5:	;Обработчик IRQ5 - прерывание контроллера жесткого диска
int 50h	
int 0Dh	
iret	
IRQ6:	;Обработчик IRQ6 - прерывание контроллера флоппи-диска
int 50h	
int 0Eh	
iret	
IRQ7:	;Обработчик IRQ7 - прерывание принтера
int 50h	
int 0Fh	
iret	
IRQ8:	;Обработчик IRQ8 - прерывание RTC
int 50h	
int 70h	
iret	
IRQ9:	;Обработчик IRQ9 - прерывание контроллера EGA
int 50h	
int 71h	
iret	
IRQ10:	;Обработчик IRQ10 - зарезервированно
int 50h	
int 72h	
iret	
IRQ11:	;Обработчик IRQ11 - зарезервировано
int 50h	
int 73h	
iret	
IRQ12:	;Обработчик IRQ12 - зарезервировано
int 50h	
int 74h	
iret	
IRQ13:	;Обработчик IRQ13 - прерывание математического сопроцессора
int 50h	
int 75h	
iret	
IRQ14:	;Обработчик IRQ14 - прерывание контроллера жесткого диска
int 50h	
int 76h	
iret	
IRQ15:	;Обработчик IRQ15 - зарезервировано
int 50h	
int 77h	
iret	
DISPLAY:	;Обработчик INT15 - вывод значения регистров контроллера прерываний
pusha	
push es	;Адрес видеопамати
mov ax,0B800h	
mov es,ax	;B es
mov cx,8	
xor di,di	
mov di,0	

```

xor bx,bx                                ;Вывод заголовка
bufferOutput:
mov al,cs:[buffer+bx]
mov es:[di],al
inc bx
inc di
mov al,BLACK
mov es:[di],al
inc di
loop bufferOutput
in al,21h                                ;Получение значения регистра масок ведущего контроллера
mov cs:[registers+0],al
in al,0A1h                                ;Получение значения регистра масок ведомого контроллера
mov cs:[registers+3],al
mov al,00001011b                          ;Команда OCW3
                                           ;0XXXXXXX - надо
                                           ;X00XXXXX - не изменять режим маскирования
                                           ;XXX01XXX - признак OCW3
                                           ;XXXXX0XX - не режим опроса
                                           ;XXXXXX11 - считать регистр обслуживающих прерываний

out 20h,al
in al,20h
mov cs:[registers+2],al
mov al,00001011b                          ;Команда OCW3
                                           ;0XXXXXXX - надо
                                           ;X00XXXXX - не изменять режим маскирования
                                           ;XXX01XXX - признак OCW3
                                           ;XXXXX0XX - не режим опроса
                                           ;XXXXXX11 - считать регистр обслуживающих прерываний

out 0A0h,al
in al,0A0h
mov cs:[registers+5],al
mov al,00001010b                          ;Команда OCW3
                                           ;0XXXXXXX - надо
                                           ;X00XXXXX - не изменять режим маскирования
                                           ;XXX01XXX - признак OCW3
                                           ;XXXXX0XX - не режим опроса
                                           ;XXXXXX11 - считать регистр запросов на прерывания

out 20h,al
in al,20h
mov cs:[registers+1],al
mov al,00001010b                          ;Команда OCW3
                                           ;0XXXXXXX - надо
                                           ;X00XXXXX - не изменять режим маскирования
                                           ;XXX01XXX - признак OCW3
                                           ;XXXXX0XX - не режим опроса
                                           ;XXXXXX11 - считать регистр запросов на прерывания

out 0A0h,al
in al,0A0h
mov cs:[registers+4],al
mov cx,6                                ;Цикл по выводу регистров через видеопамять
registersOutput:
    mov dx,cx
    mov bx,6
    sub bx,dx
    mov al,cs:[registers+bx];Получение значения регистра из памяти

```

```

    lea bx,cs:[temp]          ;Временный буфер для вывода
    push cx
    mov cx,8
    btbs_lp:                  ;Перевод числа в двоичный вид
        rol al,1              ;Циклический сдвиг al влево на 1 бит
        jc btbs_1             ;Если выдвинутый бит = 1, то переход
        mov cs:[bx],'0'       ;Добавление символа '0' в строку
        jmp btbs_end
    btbs_1:
        mov cs:[bx],'1'       ;Добавление символа '1' в строку
    btbs_end:
        inc bx                ;Инкремент di
        loop btbs_lp          ;Команда цикла
    mov cx,8
    xor bx,bx
    registerOutput:           ;Цикл по выводу буфера
        mov al,cs:[temp+bx]
        mov es:[di],al
        inc di
        inc bx
        mov al,GREEN
        mov es:[di],al
        inc di
    loop registerOutput
    mov al, ' '                ;Добавить пробел после буфера
    mov es:[di],al
    inc di
    mov al,GREEN
    mov es:[di],al
    inc di

    pop cx
    loop registersOutput
    pop es
    popa
    iret
    buffer db "IRQ: "          ;Буфер для заголовка
    temp db 8 dup ('6')        ;Буфер для обработки и вывода значения регистра
    registers db 6 dup (?)      ;Массив регистров
setInterrupts:                ;Процедура установки обработчиков аппаратных прерываний и
переинициализации контроллера
    mov ah,0h                  ;Установка видеорежима
    mov al,3h                  ;Консоль, 16 цветов, 80*25 символов,
    int 10h                    ;Установить видео режим
    push ds
    push cs
    pop ds
    mov ax,2560h                ;Инициализация векторов прерываний 60h-6Fh
    mov dx, offset IRQ0
    int 21h
    mov ax,2561h
    mov dx, offset IRQ1
    int 21h
    mov ax,2562h
    mov dx, offset IRQ2
    int 21h
    mov ax,2563h

```

```

mov dx, offset IRQ3
int 21h
mov ax,2564h
mov dx, offset IRQ4
int 21h
mov ax,2565h
mov dx, offset IRQ5
int 21h
mov ax,2566h
mov dx, offset IRQ6
int 21h
mov ax,2567h
mov dx, offset IRQ7
int 21h
mov ax,2568h
mov dx, offset IRQ8
int 21h
mov ax,2569h
mov dx, offset IRQ9
int 21h
mov ax,256Ah
mov dx, offset IRQ10
int 21h
mov ax,256Bh
mov dx, offset IRQ11
int 21h
mov ax,256Ch
mov dx, offset IRQ12
int 21h
mov ax,256Dh
mov dx, offset IRQ13
int 21h
mov ax,256Eh
mov dx, offset IRQ14
int 21h
mov ax,256Fh
mov dx, offset IRQ15
int 21h
mov ax,2550h
mov dx, offset DISPLAY
int 21h
cli
    mov al, 11h
    out 20h, al
    out 0A0h, al
    mov al, 60h
    out 21h, al
    mov al, 68h
    out 0A1h, al
    mov al, 04h
    out 21h, al
    mov al, 02h
;Запрет аппаратных прерываний
;ICW1 - начало инициализации контроллеров
;0001XXXX - признак ICW1
;XXXX0XXX - срабатывание по перепаду сигнала
;XXXXX1XX - размер вектора прерываний
;XXXXXX0X - каскадное подключение контроллеров
;XXXXXXX0 - будет послано ICW4
;Инициализация ведущего контроллера
;Инициализация ведомого контроллера
;ICW2 - базовые номера вектора прерывания ведущего контроллера
;ICW2 - базовые номера вектора прерывания ведомого контроллера
;ICW3 - ведущий контроллер подключен к 3 линии
;ICW3 - ведомый контроллер подключен к 3 линии

```

```

        out 0A1h, al
        mov al, 11h                ;ICW4 - режим специальной полной вложенности для ведущего
контроллера
        out 21h, al
        mov al, 01h                ;ICW4 - режим обычной полной вложенности для ведомого контроллера
        out 0A1h, al
        sti
        lea dx, setInterrupts      ;Оставить программу резидентной
        int 27h
        ret
    ends code

```

## 2. Тестовые пример

```

IRQ: 00000000 00000000 00000001 00000000 00000000 00000000
A:\>

```

## 3. Заключение

Написал обработчики аппаратных прерываний, которые будут установлены на используемые пользовательские прерывания. Выводил на экран в двоичной форме регистры контроллеров прерываний. Осуществил переход на стандартные обработчики прерываний.