

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №5
по дисциплине «Организация ЭВМ и систем»
ТЕМА: Разработка собственного прерывания.

Студент гр. 1303

Беззубов Д.В.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2022

Цель работы.

Разработать собственную процедуру прерывания.

Задание.

Вариант 1С.

1 – 08H - прерывание от системного таймера - генерируется автоматически операционной системой 18 раз в сек;

С – Выдача звукового сигнала с заданной длительностью звучания.

Выполнение работы

1. Объявляются два двухбайтовых сегмента памяти *KEEP_IP* и *KEEP_CS*. Они используются для сохранения смещения до оригинального прерывания и позволяют в дальнейшем восстановить исходные вектора прерывания.

2. Описывается процедура *SUBR_INT*, которая и является прерыванием, написанным нами. В данной процедуре в начале кладутся в стек все регистры, которые будут изменены (таким образом они сохраняются), затем осуществляется взаимодействие с динамиком компьютера – выставляется частота звука, время звучания, сохраняется состояние порта, биты, отвечающие за доступ к динамику и его включение выставляются в 1. После проигрывания звука порт возвращается в исходное состояние. Все сохраненные регистры вынимаются из стека, а так же обеспечивается разрешение прерываний более низкого уровня во время действия данного.

3. В главной процедуре смещение и сегмент прерывания, которое требуется заменить сохраняется в объявленные сегменты памяти:

```
MOV AH, 35H
MOV AL, 08H
INT 21H
MOV KEEP_IP, BX ; сохраняем смещение старого прерывания
MOV KEEP_CS, ES ; и сегмент
```

Рис.1 – Сохранение смещения до заменяемого прерывания 08H

Функция 35H прерывания 21H дает вектор прерывания, записанного в нижний байт регистра AX. Смещение и сегмент данного регистра записываются в регистры BX и ES, соответственно, они сохраняются в *KEEP_IP* и *KEEP_CS*.

4. Далее записывается новое прерывание. Функция 25H считывает смещение до него из DX и сегмент из DS и устанавливает его в вектор прерывания.

5. Вызывается новое прерывание, а затем восстанавливается исходный вектор прерывания.

Выводы

Разработано собственное прерывание, выдающее звук с заданной длительностью звучания и заменяющее прерывание 08H.

ПРИЛОЖЕНИЕ А

Название файла: lr5.asm

```
DATA SEGMENT
    KEEP_IP DW 0
    KEEP_CS DW 0
DATA ENDS

AStack SEGMENT STACK
    DW 1024 DUP(?)
AStack ENDS

CODE SEGMENT
    ASSUME CS:CODE, DS:DATA, SS:AStack

SUBR_INT PROC FAR
    push AX
    push BX
    push DX
    push CX

    mov di, 6000 ;freq
    mov bx, 5 ;time
    mov al, 0b6h
    out 43h, al
    mov dx, 0014h
    mov ax, 4f38h
    div di

    ; set freq
    out 42h, al
    mov al, ah
    out 42h, al

    ;sound on
    in al, 61H ;cur port state to al
    mov ah, al ;save state in ah
    or al, 3 ;set 0 and 1 bit at 1
    out 61H, al ;speaker on

l1: mov cx, 2801h
l2: loop l2
    dec bx
    jnz l1
    mov al, ah
    and al, 11111100b
    out 61H, al

    pop AX
    pop BX
    pop DX
    pop CX
    mov AL, 20H
    out 20H, AL
    iret
SUBR_INT ENDP
```

```

Main PROC FAR
    push ds
    sub ax,ax
    push ax
    mov ax, DATA
    mov ds, ax

    MOV AH, 35H
    MOV AL, 08H
    INT 21H
    MOV KEEP_IP, BX ; сохраняем смещение старого прерывания
    MOV KEEP_CS, ES ; и сегмент

    push DS
    mov DX, offset SUBR_INT
    mov ax, seg SUBR_INT
    mov ds, ax
    mov ah, 25h
    mov al, 08H
    int 21h
    pop ds

    int 08H

    CLI
    PUSH DS
    MOV DX, KEEP_IP
    MOV AX, KEEP_CS
    MOV DS, AX
    MOV AH, 25H
    MOV AL, 08H
    INT 21H ; восстанавливаем вектор
    POP DS
    STI

    ret
Main ENDP

CODE ENDS
    END Main

```