

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: «Написание собственного прерывания.»

Студентка гр. 1381

Васильева О. М.

Преподаватель

Ефремов М. А.

Санкт-Петербург

2022

Задание (Вариант 3)

Шифр задания: 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.

Исходный код программы см. в приложении А

1. В сегменте данных объявляется два двухбайтовых сегмента `KEEP_CS` и `KEEP_IP`, которые хранят сегмент и смещение до оригинального прерывания, чтобы в дальнейшем восстановить исходных вектор прерывания.

2. В головной процедуре сегмент и смещение прерывание, которое требуется заменить, устанавливаются в объявленные сегменты (сегменты, которые объявлены в пункте выше).

3. Для задания адреса собственного прерывания с заданным номером в таблицу векторов прерываний используется функция `25H` прерывания `21H`, которая устанавливает вектор прерывания на указанный адрес нового обработчика.

4. Затем в конце программы восстанавливается старый вектор прерывания.

5. Выше описанные действия происходят в головной процедуре.

6. Сама программа обработки прерывание описана в процедуре `SUBR_INT`, то есть процедура написанная в соответствии с заданием.

7. В данной процедуре по на стек кладутся изменяемые регистры, которые мы тем самым сохраняем. Затем устанавливается частота звука и длительность. В метке `start` осуществляется включение динамика – настройка высоты звука, подключения 2-ого канала для генерации звука. Также программа дает порту `61h` установку на пропуск сигнала на динамик (также сохраняется первоначальное значение кода для управляющего порта для дальнейших действий по отключению динамика). Далее осуществляется цикл длительности работы динамика. По метке `finish` происходит отключение динамика. Потом происходит восстановление регистров, помещенных на стек.

Вывод.

В ходе лабораторной работы было реализовано собственное прерывание в соответствии с условие поставленной задачи.

ПРИЛОЖЕНИЕ А. Код программ.

Имя файла: lr5.asm

DATA SEGMENT

KEEP_CS DW 0 ;буфер для хранения сегмента

KEEP_IP DW 0 ;смещение вектора прерывания

DATA ENDS

STACK SEGMENT STACK

DW 1024 DUP(?)

STACK ENDS

CODE SEGMENT

ASSUME CS:CODE, DS:DATA, SS:STACK

SUBR_INT PROC FAR

push ax ; сохранение изменяемых регистров

push bx ;

push dx ;

push cx ;

mov di, 6000 ;частота звука

mov bx, 10 ;длительность (время)

start: ;включение динамика

mov al, 10110110b ;управляющее слово таймера:канал 2

out 43h, al ;выводим значение в порт таймера

mov dx,0014H

mov ax, 1000 ;высота звука

div di ;деление al = ax/di ah=остаток

out 42h, al ;выводим младший байт счетчика во 2-й канал таймера

mov al, ah

out 42h, al ;выводим старший байт

```
in al, 61h ;текущее состояние порта 61h в AL
mov ah, al
or al, 3 ; устанавливаем биты 0 и 1 (включить спикер и использовать 2-й
канал для генерации импульсов спикера)
out 61h, al ; выводим значение в управляющий регистр
```

```
l1:mov cx, 2801h
l2:loop l2 ;цикл во время работы динамика
dec bx ;минус 1
jnz l1 ;пока не равно нулю
```

```
finish: ;выключение динамика
mov al, ah ; получаем значение из управляющего регистра порта в PPI
(текущее состояние порта 61h)
and al, 11111100b ; сбрасываем биты 0 и 1
out 61h,al ; выводим значение в
управляющий регистр порта В PPI (контроллера 8255)
```

```
pop ax ; восстановление регистров
pop bx ;
pop cx ;
pop dx ;
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
```

```
;Запоминание текущего вектора прерывания
mov ah, 35h ;функция получения вектора
```

```

mov al, 08h ;номер вектор
int 21h
mov KEEP_IP, bx ;запоминание смещения
mov KEEP_CS, es ;и сегмента

;Установка вектора прерывания
push ds
mov dx, OFFSET SUBR_INT ;смещение для процедуры в DX
mov ax, SEG SUBR_INT ;сегмент процедуры
mov ds, ax ;помещаем в DS
mov ah, 25h ;функция установки вектора
mov al, 08h ;номер вектора
int 21h ;меняем прерывание
pop ds
int 08h ;вызов прерывания

;Восстановление изначального вектора прерывания
CLI
push ds
mov dx, KEEP_IP ;смещение для процедуры в DX
mov ax, KEEP_CS ;сегмент процедуры
mov ds, ax ;помещаем в DS
mov ah, 25h ;функция установки вектора
mov al, 08h ;номер вектора
int 21h ;восстанавливаем старый вектор
pop ds
STI
RET

Main      ENDP

CODE ENDS
END Main

```