

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

ОТЧЕТ
по лабораторной работе №5
по дисциплине «ОЭВМиС»
Тема: "Разработка собственного прерывания "

Студент гр. 9383

Крейсманн К.В.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2020

Цель работы.

Получить навыки создания собственных прерываний.

Задание.

Разработать собственное прерывание. Шифр: 3А.

3 - 23h - прерывание, генерируемое при нажатии клавиш Control+c.

А - печать сообщения на экране

Ход работы:

В сегменте данных создаются переменные типа DW для хранения сегмента и смещения прерывания KEEP_CS и KEEP_IP. Также создается строка, которая будет выводиться во время обработки прерывания message.

В сегменте кода создана процедура обработки прерывания MY_INT, которая выводит строку, message на экран.

Прерывание меняется следующим образом: с помощью функции 35h прерывания 21h получаем вектор прерывания 23h. Затем запоминаем этот вектор в переменные KEEP_CS и KEEP_IP. Далее с помощью функции 25h прерывания 21h меняем прерывание 23h.

В конце программы восстанавливаем вектор прерывания.

Разработанный программный код см. в приложении А.

Выводы.

Получены навыки создания собственных прерываний.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Файл lb5.asm

```
dosseg
.model small
.stack 400h
.data
    KEEP_CS DW 0 ;для хранения сегмента
    KEEP_IP DW 0 ;для хранения смещения вектора прерывания
    message db 'Control + c$'
.code
    mov ax,@data
    mov ds,ax
    mov ah,35h ;функция получения вектора
    mov al,23h ;номер прерывания
    int 21h ;получаем вектор
    mov keep_cs,es ;запоминание сегмента вектора прерывания
    mov keep_ip,bx ;запоминание смещения
    push ds ;сохраняем ds
    mov ax, seg MY_INT ;сегмент процедуры помещаем в ax
    mov ds,ax ;перемещаем в ds
    mov dx,offset MY_INT ;смещение для процедуры помещаем в dx
    mov ah,25h ;функция установки вектора
    mov al,23h ;номер прерывания
    int 21h ;меняем прерывание
    pop ds
    jmp metka

MY_INT proc far
    push dx
    push ax
    mov dx,offset message ;смещение для сообщения
```

```

mov ah,9h          ;номер функции
int 21h            ;вызов прерывания
mov al,20h         ;>разрешение обработки прерываний
out 20h,al         ;>более низкого уровня
pop ax
pop dx
iret
MY_INT endp
metka:
; int 23h
push ds
mov dx,keep_ip     ;восстанавливаем смещение для прерывания
mov ax,keep_cs     ;восстанавливаем сегмент прерывания
mov ds,ax          ;заносим его в ds
mov ah,25h         ;функция установки вектора
mov al,23h         ;номер прерывания
int 21h            ;меняем прерывание
pop ds
mov ah,4ch
int 21h
end

```

Файл lb5.lst

#Microsoft (R) Macro Assembler Version 5.00

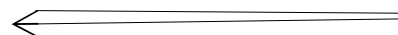
11/4/20 17:51:03

Page 1-1

```

dosseg
.model small
0400 .stack 400h
0000 .data
0000 0000 KEEP_CS DW 0

```



0002 0000	KEEP_IP DW 0
0004 43 6F 6E 74 72 6F 6C	message db 'Control + c\$'
20 2B 20 63 24	
0000	.code
0000 B8 ---- R	mov ax,@data
0003 8E D8	mov ds,ax
0005 B4 35	mov ah,35h
0007 B0 23	mov al,23h
0009 CD 21	int 21h
000B 8C 06 0000 R	mov keep_cs,es
000F 89 1E 0002 R	mov keep_ip,bx
0013 1E	push ds
0014 B8 ---- R	mov ax, seg MY_INT
0017 8E D8	mov ds,ax
0019 BA 0026 R	mov dx,offset MY_INT
001C B4 25	mov ah,25h
001E B0 23	mov al,23h
0020 CD 21	int 21h
0022 1F	pop ds
0023 EB 11 90	jmp metka
0026	MY_INT proc far
0026 52	push dx
0027 50	push ax
0028 BA 0004 R	mov dx,offset message
002B B4 09	mov ah,9h
002D CD 21	int 21h
002F B0 20	mov al,20h

```
0031 E6 20          out 20h,al
0033 58             pop ax
0034 5A             pop dx
0035 CF             iret
0036               MY_INT endp
0036               metka:
                   ; int 23h
0036 1E             push ds
0037 8B 16 0002 R    mov dx,keep_ip
003B A1 0000 R      mov ax,keep_cs
003E 8E D8          mov ds,ax
0040 B4 25          mov ah,25h
0042 B0 23          mov al,23h
0044 CD 21          int 21h
0046 1F             pop ds
0047 B4 4C          mov ah,4ch
0049 CD 21          int 21h
004B               end
```

Segments and Groups:

N a m e	Length	Align	Combine	Class
DGROUP				GROUP

```

_DATA ..... 0010 WORD    PUBLIC  'DATA'
STACK ..... 0400 PARA STACK  'STACK'
_TEXT ..... 004B WORD    PUBLIC  'CODE'

```

Symbols:

N a m e	Type	Value	Attr
KEEP_CS	L WORD	0000	_DATA
KEEP_IP	L WORD	0002	_DATA
MESSAGE	L BYTE	0004	_DATA
METKA	L NEAR	0036	_TEXT
MY_INT	F PROC	0026	_TEXT Length = 0010
@CODE	TEXT	_TEXT	
@CODESIZE	TEXT	0	
@DATASIZE	TEXT	0	
@FILENAME	TEXT	lb5	

50 Source Lines

50 Total Lines

19 Symbols

51182 + 465362 Bytes symbol space free

0 Warning Errors

0 Severe Errors