

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра математического обеспечения и применения ЭВМ**

**ОТЧЕТ**  
**по лабораторной работе № 5**  
**по дисциплине «Операционные системы»**  
**Тема: Сопряжение стандартного и пользовательского обработчиков**  
**прерываний.**

Студент гр. 0382

Диденко Д. В.

Преподаватель

Ефремов М. А.

Санкт-Петербург

2022

### **Цель работы.**

Исследование возможности встраивания пользовательского обработчика прерываний в стандартный обработчик от клавиатуры. Пользовательский обработчик прерывания получает управление по прерыванию (int 09h) при нажатии клавиши на клавиатуре. Он обрабатывает скан-код и осуществляет определенные действия, если скан-код совпадает с определенными кодами, которые он должен обрабатывать. Если скан-код не совпадает с этими кодами, то управление передаётся стандартному прерыванию

### **Постановка задачи.**

**Шаг 1.** Для выполнения лабораторной работы необходимо написать и отладить программный модуль типа .EXE, который выполняет такие же функции, как в программе ЛР 4, а именно:

Программа должна содержать код устанавливаемого прерывания в виде удаленной процедуры. Этот код будет работать после установки при возникновении прерывания. Он должен выполнять следующие функции:

- Сохранить значения регистров в стеке при входе и восстановить их при выходе.
- При выполнении тела процедуры анализируется скан-код.
- Если этот код совпадает с одним из заданных, то требуемый код записывается в буфер клавиатуры.
- Если этот код не совпадает ни с одним из заданных, то осуществляется передача управления стандартному обработчику прерывания.

**Шаг 2.** Запустите отлаженную программу и убедитесь, что резидентный обработчик прерывания 09h установлен. Работа прерывания проверяется введением различных символов, обрабатываемых установленным обработчиком и стандартным обработчиком.

**Шаг 3.** Также необходимо проверить размещение прерывания в памяти. Для этого запустите программу ЛР 3, которая отображает карту памяти в виде списка блоков МСВ. Полученные результаты поместите в отчет.

**Шаг 4.** Запустите отлаженную программу еще раз и убедитесь, что программа определяет установленный обработчик прерываний. Полученные результаты поместите в отчет.

**Шаг 5.** Запустите отлаженную программу с ключом выгрузки и убедитесь, что резидентный обработчик прерывания выгружен, то есть сообщения на экран не выводятся, а память, занятая резидентом освобождена. Для этого также следует запустить программу ЛР 3. Полученные результаты поместите в отчет.

### **Выполнение работы.**

Были написаны строки для вывода информации:

- STR\_IST\_LOAD DB 'Itterapt is not load', 0AH, 0DH,'\$'
- STR\_ALR\_LOAD DB 'Itterapt is already loaded', 0AH, 0DH,'\$'
- STR\_SUC\_LOAD DB 'Itterapt has been loaded', 0AH, 0DH,'\$'
- STR\_IS\_UNLOAD DB 'Itterapt is unloaded', 0AH, 0DH,'\$'

Переменные для хранения флагов:

- flag db 0 – флаг удаления;
- flag\_load db 0 – флаг загрузки.

Переменные, хранящиеся в прерывании:

- PSP dw ? – сохранение адреса PSP;
- KEEP\_IP dw 0 – сохранение данных исходного прерывания;
- KEEP\_CS dw 0 – сохранение данных исходного прерывания;
- ITERRUPT\_ID dw 8f17h – уникальный идентификатор прерывания;

- STR\_INTERRUPT db SUCCESS WORK LAB! '\$' – строка вывода в прерывании;
- REQ\_KEY db 3Bh - скан код клавиши f1;
- STR\_INDEX db 0 – счётчик для вывода строки
- KEEP\_SS dw ? – для работы стека прерывания;
- KEEP\_SP dw ? – для работы стека прерывания;
- KEEP\_AX dw ? – для работы стека прерывания;
- INTERRUPT\_STACK dw 32 dup (?) – стек прерывания;
- END\_IT\_STACK dw ? – конец стека прерывания;

Были составлены функции (см. табл.1)

Таблица 1 – функции в программе

Процедура	Описание
MY_INTERRUPT	Резидентное прерывание, которое загружается в память и выполняет вывод символа от сообщения при нажатии на f1
WRITE_STRING	Вывод строки на экран
LOAD_FLAG	Проверка на наличия флага “/un”
IS_LOAD	Проверка на загрузку пользовательского прерывания в память
LOAD_INTERRUPT	Сохранение первоначального прерывания и загрузка пользовательского прерывания в память
UNLOAD_INTERRUPT	Выгрузка пользовательского прерывания из памяти, а также освобождение памяти и восстановление первоначальных прерываний
MAIN	Главная функция

В результате выполнения были получены следующие значения(рис.1-3):

```

C:\>LAB5.EXE
Iterrapt has been loaded

C:\>SUCCESS WORK LAB!_

```

Рисунок 1 – прерывание загружено в память и пример вывода для нескольких нажатий f1

```

C:\>LAB5.EXE
Iterrapt has been loaded

C:\>SUCCESS WORK LAB!
Illegal command: SUCCESS.

C:\>LAB3_2.COM

Size of accessed memory: 648240 byte
Size of extended memory: 245760 byte
MCB:01 Address: 016F PSP address: 0008 Size: 16 SD/SC:
MCB:02 Address: 0171 PSP address: 0000 Size: 64 SD/SC:
MCB:03 Address: 0176 PSP address: 0040 Size: 256 SD/SC:
MCB:04 Address: 0187 PSP address: 0192 Size: 144 SD/SC:
MCB:05 Address: 0191 PSP address: 0192 Size: 496 SD/SC: LAB5
MCB:06 Address: 01B1 PSP address: 01BC Size: 144 SD/SC:
MCB:07 Address: 01BB PSP address: 01BC Size: 832 SD/SC: LAB3_2
MCB:08 Address: 01F0 PSP address: 0000 Size: 647392 SD/SC:
C:\>_

```

Рисунок 2 – проверка на расположение в памяти

```

Size of accessed memory: 648240 byte
Size of extended memory: 245760 byte
MCB:01 Address: 016F PSP address: 0008 Size: 16 SD/SC:
MCB:02 Address: 0171 PSP address: 0000 Size: 64 SD/SC:
MCB:03 Address: 0176 PSP address: 0040 Size: 256 SD/SC:
MCB:04 Address: 0187 PSP address: 0192 Size: 144 SD/SC:
MCB:05 Address: 0191 PSP address: 0192 Size: 496 SD/SC: LAB5
MCB:06 Address: 01B1 PSP address: 01BC Size: 144 SD/SC:
MCB:07 Address: 01BB PSP address: 01BC Size: 832 SD/SC: LAB3_2
MCB:08 Address: 01F0 PSP address: 0000 Size: 647392 SD/SC:
C:\>LAB5.EXE /un
Iterrapt is unloaded

C:\>LAB3_2.COM

Size of accessed memory: 648912 byte
Size of extended memory: 245760 byte
MCB:01 Address: 016F PSP address: 0008 Size: 16 SD/SC:
MCB:02 Address: 0171 PSP address: 0000 Size: 64 SD/SC:
MCB:03 Address: 0176 PSP address: 0040 Size: 256 SD/SC:
MCB:04 Address: 0187 PSP address: 0192 Size: 144 SD/SC:
MCB:05 Address: 0191 PSP address: 0192 Size: 832 SD/SC: LAB3_2
MCB:06 Address: 01C6 PSP address: 0000 Size: 648064 SD/SC:
C:\>_

```

Рисунок 3 – выгрузка прерывания

### **Выводы.**

В ходе лабораторной работы были исследованы возможности встраивания пользовательского обработчика прерываний в стандартный обработчик от клавиатуры. Загружается и выгружается резидент, а также производится проверка флагов и загрузки прерывание в память. С помощью rout при нажатии на клавишу f1 на экран посимвольно выводится строка, определённая в этом прерывании.

## **ПРИЛОЖЕНИЕ А**

### **ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Какого типа прерывания использовались в работе?

В данной работе использовались прерывания функции DOS – int 21h и прерывания функции BIOS.

2. Чем отличается скан код от кода ASCII?

Скан-код – уникальное число-идентификатор клавиши, используется для определения нажатой клавиши контроллером клавиатуры. Контроллер пересылает скан-код в порт.

Код ASCII – это уникальный код для каждого символа.

Скан код характеризует клавишу, а код ANCSII – символ.