

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

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

Студент гр. 0381

Ефимов Н.Д.

Преподаватель

Губкин А.Ф.

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

Задание.

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 'Interrupt is not loaded', 0AH, 0DH, '$'
```

```
STR_ALR_LOAD DB 'Interrupt was already loaded', 0AH, 0DH, '$'
```

```
STR_SUC_LOAD DB 'Interrupt is loaded', 0AH, 0DH, '$'
```

```
STR_IS_UNLOAD DB 'Interrupt 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_ITERRUPT 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;

ITERRUPT_STACK dw 32 dup (?) – стек прерывания;

END_IT_STACK dw ? – конец стека прерывания;

Были написаны ряд процедур для работы программы: CUSTOM_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. Загрузим прерывание в память и, используя 3 лабораторную работу, проверим, куда она сохраняется.

```
F:\>lab5
Interrupt is loaded

F:\>lab3

Size of accessed memory: 648240 byte
Size of extended memory: 0 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
MCB:08 Address: 01F0 PSP address: 0000 Size: 647392 SD/SC: 1|Uïwâw◆
F:\>
```

Дальше выгрузим и еще раз проверим, что случилось с памятью.

```
F:\>lab3.com

Size of accessed memory: 648912 byte
Size of extended memory: 0 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
MCB:06 Address: 01C6 PSP address: 0000 Size: 648064 SD/SC:
F:\>
```

Выводы.

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

ПРИЛОЖЕНИЕ А

ОТВЕТЫ НА 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 ASCII – символ.

ПРИЛОЖЕНИЕ Б

ТЕСТИРОВАНИЕ

Если результаты тестирования велики (больше 1 страницы), то их выносят в приложение.

Процесс тестирования можно представить в виде таблицы, например:

Таблица Б.2 - Примеры тестовых случаев

№ п/п	Входные данные	Выходные данные	Комментарии
1.			
2.			
3.			
...			

Обратите внимание, что в нумерации таблицы в приложении обязательно должен быть в качестве префикса номер самого приложения: А.