

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

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

Студент гр.

Кирильцев Д. А.

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

Постановка задачи.

Шаг 1. Для выполнения лабораторной работы необходимо написать и отладить программный модуль типа .EXE, который выполняет такие же процедуры, 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 процедуры:

- 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:

- `int_not_loaded db 'Interrupt is not loaded.', 10, 13, '$'`
- `int_unloaded db 'Interrupt unloaded successfully.', 10, 13, '$'`
- `int_loaded db 'Interrupt loaded successfully.', 10, 13, '$'`
- `int_in_use db 'Interrupt is loaded already.', 10, 13, '$'`

Переменные для хранения флагов:

- `unload_key db 0`
- `in_use db 0`

Переменные, хранящиеся в прерывании:

- `int_id dw 8f17h`
- `psp dw ?`
- `keep_ip dw 0`
- `keep_cs dw 0`
- `keep_ss dw ?`

- keep_sp dw ?
- keep_ax dw ?
- msg db 'ASU is USA backwards!' , '\$'
- i db 0
- key db 2Ah ; left shift
- int_stack dw 32 dup (?)
- int_stack_end dw ?

Были составлены процедуры (см. табл.1)

Таблица 1 – процедуры в программе

Процедура	Описание
custom_interruption	Резидентное прерывание, которое загружается в память и выполняет вывод символа от сообщения при нажатии на левый shift
printer	Вывод строки на экран
unload_key_check	Проверка на наличия ключа “/up”
is_in_use	Проверка, загружено ли прерывание уже.
int_load	Сохранение первоначального прерывания и загрузка пользовательского прерывания в память
int_unload	Выгрузка пользовательского прерывания из памяти, а также освобождение памяти и восстановление первоначальных прерываний
Main	Главная процедура

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

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX -
E:\OS\UTILIT~1>lab5
Interruption loaded successfully.
E:\OS\UTILIT~1>ASU is USA backwards!ASU is USA backwards!ASU is _
```

Рисунок 1 – прерывание успешно загружено в память, после чего shift (левый) удерживался в течение 4 секунд

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX -
E:\OS\UTILIT~1>lab5
Interruption loaded successfully.
E:\OS\UTILIT~1>lab32.com
AMS: 647968 byte(s)
EMS: 245760 byte(s)
MCB:0 1 Address: 016F PSP: 0008 Paragraphs: 16 System Data/Code:
MCB:0 2 Address: 0171 PSP: 0000 Paragraphs: 64 System Data/Code:
MCB:0 3 Address: 0176 PSP: 0040 Paragraphs: 256 System Data/Code:
MCB:0 4 Address: 0187 PSP: 0192 Paragraphs: 144 System Data/Code:
MCB:0 5 Address: 0191 PSP: 0192 Paragraphs: 768 System Data/Code: LAB5
MCB:0 6 Address: 01C2 PSP: 01CD Paragraphs: 144 System Data/Code:
MCB:0 7 Address: 01CC PSP: 01CD Paragraphs: 816 System Data/Code: LAB32
MCB:0 8 Address: 0200 PSP: 0000 Paragraphs: 647136 System Data/Code: >H 8tJr
E:\OS\UTILIT~1>
```

Рисунок 2 – проверка расположения в памяти

```

E:\OS\UTILIT~1>lab5 /un
Interruption unloaded successfully.

E:\OS\UTILIT~1>lab32.com

AMS: 648912 byte(s)
EMS: 245760 byte(s)

MCB:0 1 Adress: 016F PSP: 0008 Paragraphs: 16 System Data/Code:
MCB:0 2 Adress: 0171 PSP: 0000 Paragraphs: 64 System Data/Code:
MCB:0 3 Adress: 0176 PSP: 0040 Paragraphs: 256 System Data/Code:
MCB:0 4 Adress: 0187 PSP: 0192 Paragraphs: 144 System Data/Code:
MCB:0 5 Adress: 0191 PSP: 0192 Paragraphs: 816 System Data/Code: LAB32
MCB:0 6 Adress: 01C5 PSP: 0000 Paragraphs: 648080 System Data/Code: A220 17

E:\OS\UTILIT~1>

```

Рисунок 3 – выгрузка прерывания

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX -
E:\OS\UTILIT~1>lab5
Interruption loaded successfully.

E:\OS\UTILIT~1>lab5
Interruption is loaded already.

E:\OS\UTILIT~1>lab5 /un
Interruption unloaded successfully.

E:\OS\UTILIT~1>lab5 /un
Interruption is not loaded.

E:\OS\UTILIT~1>

```

Рисунок 4 — Невозможность загрузки в то время, как прерывание загружено, и невозможность выгрузки в то время, как прерывание выгружено.

Выводы.

В ходе лабораторной работы было изучено прерывание от клавиатуры, а также механизм написания пользовательского прерывания.

ПРИЛОЖЕНИЕ А

ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Какого типа прерывания использовались в работе?

прерывания процедуры DOS – int 21h

прерывания процедуры 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.

ASCII – уникальный код для каждого символа.

Скан код соответствует клавише, а код ASCII – символу.