

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

Студентка гр. 0381

Ионина К.С.

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 передается 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:

- 1) Проверяет, установлено ли пользовательское прерывание с вектором 09h.
- 2) Если прерывание не установлено то, устанавливает резидентную функцию для обработки прерывания и настраивает вектор прерываний. Адрес точки входа в стандартный обработчик прерывания находится в теле пользовательского обработчика. Осуществляется выход по функции 4Ch прерывания int 21h.
- 3) Если прерывание установлено, то выводится соответствующее сообщение и осуществляется выход по функции 4Ch прерывания int 21h. Выгрузка прерывания по соответствующему значению параметра в командной строке /un. Выгрузка прерывания состоит в восстановлении стандартного вектора прерываний и освобождении памяти, занимаемой резидентом. Затем осуществляется выход по функции 4Ch прерывания int 21h. Для того чтобы проверить установку прерывания, можно поступить следующим образом. Прочитать адрес, записанный в векторе прерывания. Предположим, что этот адрес указывает на точку входа в установленный резидент. На определенном, известном смещении в теле резидента располагается сигнатура, некоторый код, который идентифицирует резидент. Сравнив известное значение сигнатуры с реальным кодом, находящимся в резиденте, можно определить, установлен ли

резидент. Если значения совпадают, то резидент установлен. Длину кода сигнатуры должна быть достаточной, чтобы сделать случайное совпадение маловероятным. Программа должна содержать код устанавливаемого прерывания в виде удаленной процедуры. Этот код будет работать после установки при возникновении прерывания. Он должен выполнять следующие функции:

- 1) Сохранить значения регистров в стеке при входе и восстановить их при выходе.
- 2) При выполнении тела процедуры анализируется скан-код.
- 3) Если этот код совпадает с одним из заданных, то требуемый код записывается в буфер клавиатуры.
- 4) 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.

Шаг 6. Ответьте на контрольные вопросы.

Выполнение работы.

Строки для вывода информации:

```
STR_NOMARKER db " /un",0dh
```

```
STR_LOADING db "Interrupt is loaded",0Dh,0Ah,'$'
```

```
STR_IS_LOADING db "Interrupt already loaded",0Dh,0Ah,'$'
```

```
STR_UNLOADING db "Interrupt is unloaded",0Dh,0Ah,'$'
```

```
STR_NO_LOADING db "Interrupt isn't loaded",0Dh,0Ah,'$'
```

Шаг 1.

Процедура IS_LOADING при запуске программы осуществляет проверку на наличие установленного пользовательского прерывания. Затем с помощью условных переходов, определяется необходимое действие.

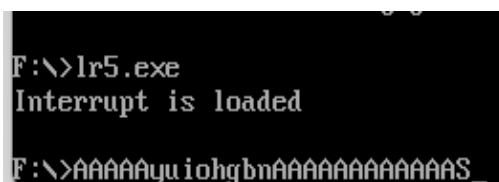
Процедура LOADING устанавливает пользовательское прерывание, используя 25h int 21h, и выходит из программы, оставляя её резидентной.

Процедура UNLOADING обращается к сохранённым в теле прерывания переменным и восстанавливает по ним стандартный код прерывания.

Процедура INTERRUPTION была взята из прошлой лабораторной работы, но претерпела изменения. Была добавлена переменная, содержащая скан-код. Если скан-код совпадает со считанным с клавиатуры, то управление передается стандартному обработчику прерывания, иначе происходит обработка прерывания с клавиатуры. После обработки прерывания клавиатуры, с помощью прерывания 16h, в буфер клавиатуры записывается заданный символ «А». В случае, когда буфер заполнен, происходит его очистка и попытка вновь произвести запись.

Шаг 2.

Резидентный обработчик прерывания установлен. При нажатии F1 выводится заданный символ «А», при нажатии других клавиш, выполняется их привычная функция.

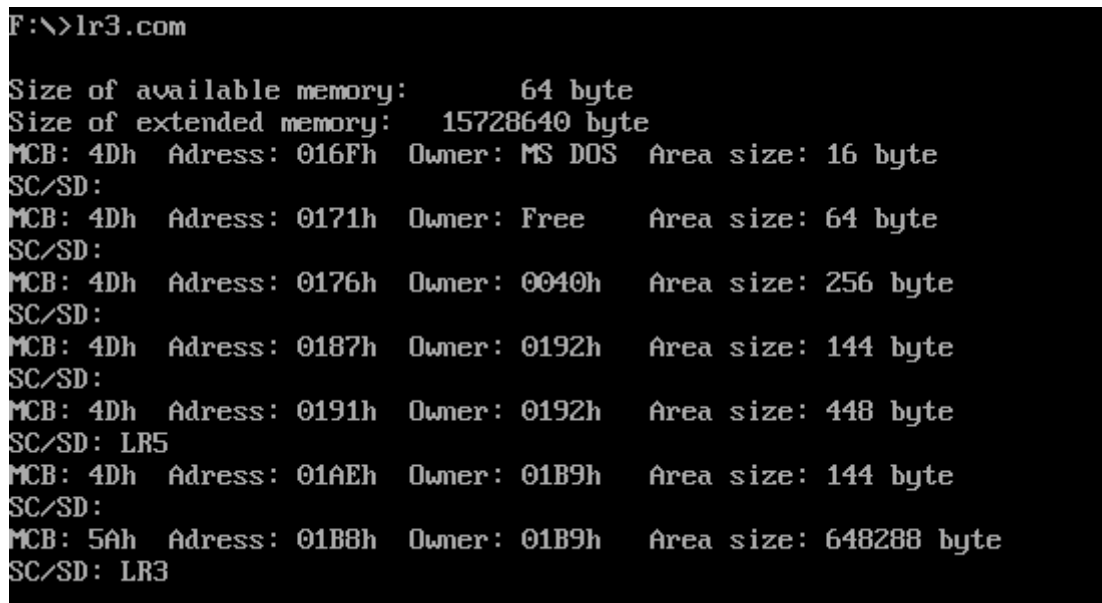


```
F:\>lr5.exe
Interrupt is loaded
F:\>AAAAAyuiohgbnAAAAAAS_
```

Рисунок 1. Запуск отлаженной программы.

Шаг 3.

Прерывание размещается в памяти по адресу 0192h. На рис.2 отображена карта памяти в виде списка блоков MCB.



```
F:\>lr3.com
Size of available memory:      64 byte
Size of extended memory: 15728640 byte
MCB: 4Dh Address: 016Fh Owner: MS DOS Area size: 16 byte
SC/SD:
MCB: 4Dh Address: 0171h Owner: Free Area size: 64 byte
SC/SD:
MCB: 4Dh Address: 0176h Owner: 0040h Area size: 256 byte
SC/SD:
MCB: 4Dh Address: 0187h Owner: 0192h Area size: 144 byte
SC/SD:
MCB: 4Dh Address: 0191h Owner: 0192h Area size: 448 byte
SC/SD: LR5
MCB: 4Dh Address: 01AEh Owner: 01B9h Area size: 144 byte
SC/SD:
MCB: 5Ah Address: 01B8h Owner: 01B9h Area size: 648288 byte
SC/SD: LR3
```

Рисунок 2. Проверка размещения прерывания в памяти.

Шаг 4.

При повторном запуске программа определяет установленный обработчик прерываний.



```
F:\>lr5.exe
Interrupt already loaded
```

Рисунок 3. Повторный вызов lr5.exe.

Шаг 5.

При запуске программы с ключом выгрузки сообщения на экран не выводятся, а память, занятая резидентом освобождена.

```
F:\>lr5.exe /un
Interrupt is unloaded

F:\>lr3.com

Size of available memory:      64 byte
Size of extended memory: 15728640 byte
MCB: 4Dh  Address: 016Fh  Owner: MS DOS  Area size: 16 byte
SC/SD:
MCB: 4Dh  Address: 0171h  Owner: Free    Area size: 64 byte
SC/SD:
MCB: 4Dh  Address: 0176h  Owner: 0040h   Area size: 256 byte
SC/SD:
MCB: 4Dh  Address: 0187h  Owner: 0192h   Area size: 144 byte
SC/SD:
MCB: 5Ah  Address: 0191h  Owner: 0192h   Area size: 648912 byte
SC/SD: LR3
```

Рисунок 4. Запуск отлаженной программы с ключом выгрузки.

Вывод.

В ходе выполнения лабораторной работы были исследованы возможности встраивания пользовательского обработчика прерываний в стандартный обработчик от клавиатуры. Пользовательский обработчик 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 передается стандартному прерыванию.

Контрольные вопросы по лабораторной работе №4.

1) Какого типа прерывания использовались в работе?

В работе были использованы программные прерывания int21h – сервис DOS, int 16h – функции BIOS и int9h – обработчик прерывания от клавиатуры.

2) Чем отличается скан код от кода ASCII?

Скан-код – это уникальный идентификатор клавиши на клавиатуре. Каждой клавише соответствует свой скан-код.

Код ASCII – идентификатор символа. Не для всех клавиш на клавиатуре существует отдельный символ. Для некоторых клавиш идентификатор символа зависит от раскладки.

Таким образом, скан-код характеризует клавишу, а код ASCII - символ.