

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

ОТЧЕТ
по лабораторной работе №4
по дисциплине «Операционные системы»
Тема:
Обработка стандартных прерываний

Студент гр. 0381

Михайлов В.А.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2022

Цель работы.

В архитектуре компьютера существуют стандартные прерывания, за которыми закреплены определенные вектора прерываний. Вектор прерываний хранит адрес подпрограммы обработчика прерываний. При возникновении прерывания, аппаратура компьютера передает управление по соответствующему адресу вектора прерывания. Обработчик прерываний получает управление и выполняет соответствующие действия.

Постановка задачи.

Шаг 1. Для выполнения лабораторной работы необходимо написать и отладить программный модуль типа .EXE, который выполняет следующие функции:

- 1) Проверяет, установлено ли пользовательское прерывание с вектором 1Ch.
- 2) Устанавливает резидентную функцию для обработки прерывания и настраивает вектор прерываний, если прерывание не установлено, и 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.
- 4) 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) Сохраняет стек прерванной программы (регистры SS и SP) в рабочих переменных и восстановить при выходе.

2) Организовать свой стек.

3) Сохранить значения регистров в стеке при входе и восстановить их при выходе.

4) При выполнении тела процедуры накапливать общее суммарное число прерываний и выводить на экран. Для вывода на экран следует использовать прерывание `int 10h`, которое позволяет непосредственно выводить информацию на экран.

5) Функция прерывания должна содержать только переменные, которые она использует.

Шаг 2. Запустите отлаженную программу и убедитесь, что резидентный обработчик прерывания `1Ch` установлен. Работа прерывания должна отображаться на экране, а 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 3. Запустите отлаженную программу еще раз и убедитесь, что программа определяет установленный обработчик прерываний. Полученные результаты поместите в отчет.

Шаг 4. 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 5. Ответьте на контрольные вопросы.

Выполнение работы.

При запуске программы вызываются процедуры CHECK_USERSET и GET_TAIL.

Процедура CHECK_USERSET реализована для того, чтобы проверить установлен ли пользовательский обработчик прерываний. Внутри неё в регистр AX записывается значение слова SIGNATURE, и производится сравнение с нужным значением. Если установлено совпадение, то в регистр AL заносится 1, в противном случае – 0.

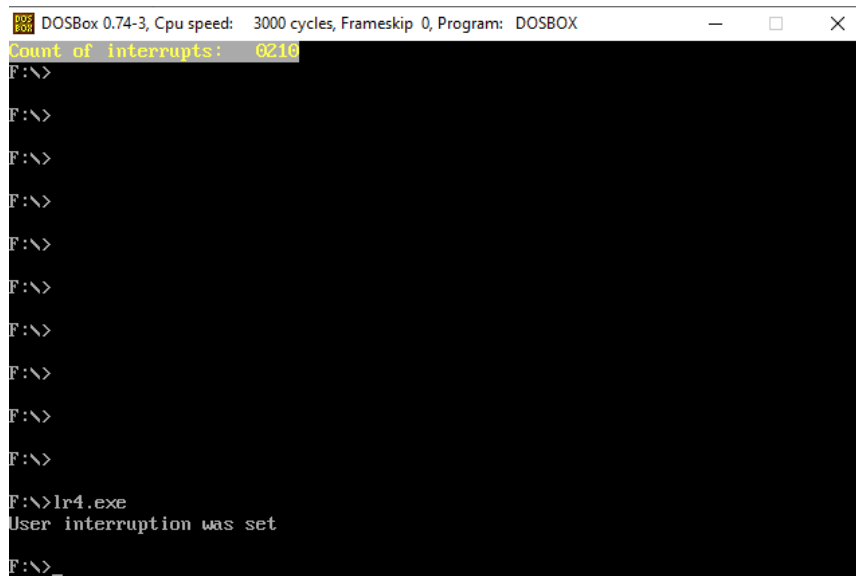
Процедура DISABLE_USERSET восстанавливает стандартное прерывание. Она вызывается, если модуль запущен с параметром /un. Наличие же этого параметра в «хвосте» запускающей команды проверяет процедура GET_TAIL на основе значения, лежащего со смещением 81h в PSP.

Если пользовательское прерывание ещё не установлено, то вызывается процедура SET_INTERRUPTION. В ней хранится адрес сегмента и смещение стандартного прерывания, чтобы его можно было восстановить, а на его место устанавливается резидентная функция. В итоге выводится сообщение «User interruption was set». Если пользовательское прерывание уже установлено, то проверяется наличие параметра /un, выводится сообщение «User interruption has already set» и, если параметр есть, то ещё выводится «Interruption restored».

Также написана процедура UPDATE_COUNTER, которая обновляет счётчик количества прерываний в строке COUNT_STR.

Тестирование.

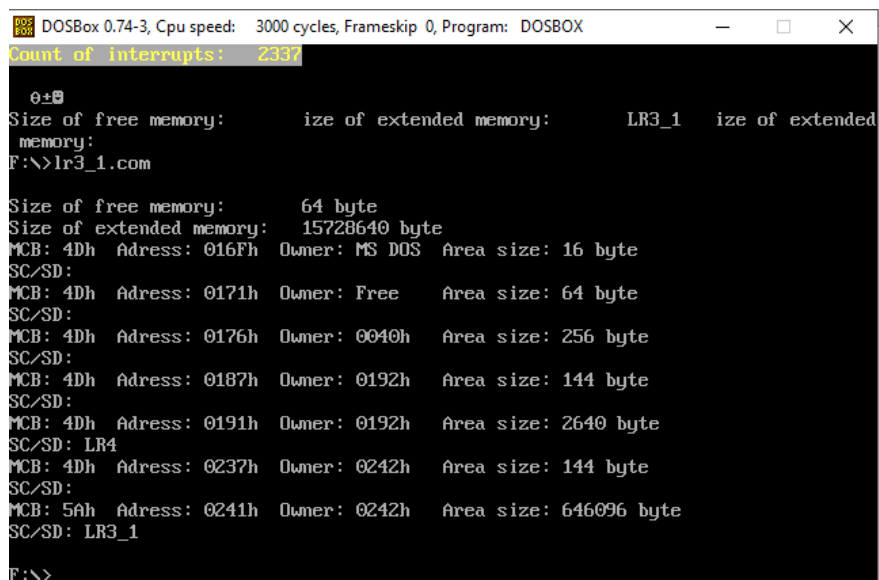
При первом вызове программа устанавливает пользовательское прерывание:



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX
Count of interrupts: 0210
F:\>
F:\>
F:\>
F:\>
F:\>
F:\>
F:\>
F:\>
F:\>
F:\>
F:\>
F:\>
F:\>lr4.exe
User interruption was set
F:\>_
```

Рис.1 – Вызов lr4.exe №1

Если вызвать программу из лаб. работы №3, то счётчик будет продолжать работать, а также будет видно, что текущая программа загружена в память:

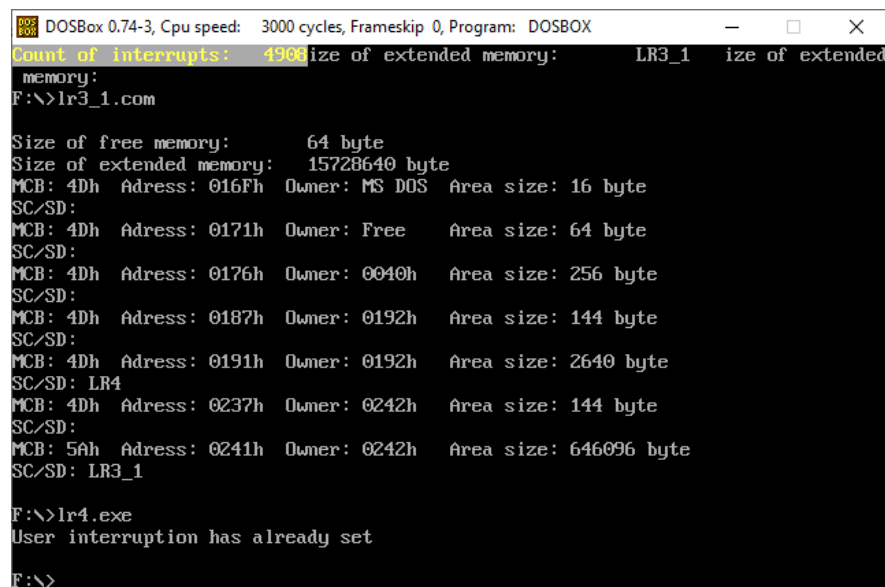


```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX
Count of interrupts: 2337
0±8
Size of free memory:      ize of extended memory:      LR3_1      ize of extended
memory:
F:\>lr3_1.com

Size of free 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: 2640 byte
SC/SD: LR4
MCB: 4Dh  Address: 0237h  Owner: 0242h  Area size: 144 byte
SC/SD:
MCB: 5Ah  Address: 0241h  Owner: 0242h  Area size: 646096 byte
SC/SD: LR3_1
F:\>
```

Рис. 2 – Вызов №1 lr3_1.com

При повторном вызове модуля lr4.exe обработчик уже установлен:



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX
Count of interrupts: 4968 size of extended memory: LR3_1 size of extended
memory:
F:\>lr3_1.com

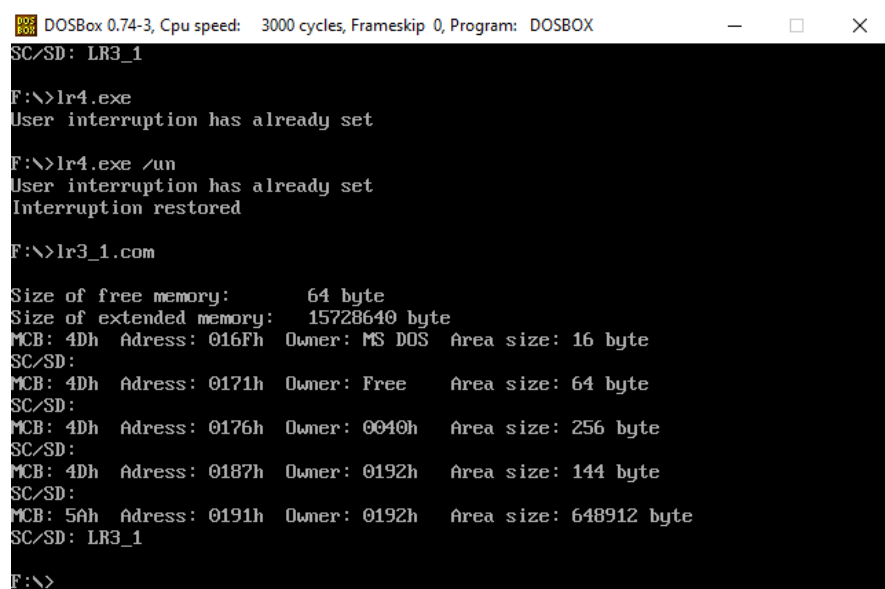
Size of free memory:      64 byte
Size of extended memory: 15728640 byte
MCB: 4Dh  Address: 016Fh  Owner: MS DOS  Area size: 16 byte
SC/SD:
MCB: 4Dh  Address: 0171h  Owner: Free    Area size: 64 byte
SC/SD:
MCB: 4Dh  Address: 0176h  Owner: 0040h   Area size: 256 byte
SC/SD:
MCB: 4Dh  Address: 0187h  Owner: 0192h   Area size: 144 byte
SC/SD:
MCB: 4Dh  Address: 0191h  Owner: 0192h   Area size: 2640 byte
SC/SD: LR4
MCB: 4Dh  Address: 0237h  Owner: 0242h   Area size: 144 byte
SC/SD:
MCB: 5Ah  Address: 0241h  Owner: 0242h   Area size: 646096 byte
SC/SD: LR3_1

F:\>lr4.exe
User interruption has already set

F:\>
```

Рис. 3 – Вызов lr4.exe №2

При запуске программы с параметром /un появится сообщение о восстановлении стандартного прерывания. Также при выполнении модуля lr3_1.com можно увидеть, что произошло освобождение памяти, занимаемой ранее резидентом:



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX
SC/SD: LR3_1

F:\>lr4.exe
User interruption has already set

F:\>lr4.exe /un
User interruption has already set
Interruption restored

F:\>lr3_1.com

Size of free 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_1

F:\>
```

Рис 4. – Вызов lr4.exe №3 + Вызов lr3_1.com № 2

Вывод.

Был построен обработчик прерываний сигналов таймера.

Контрольные вопросы.

1) Как реализован механизм прерывания от часов?

Принимается сигнал прерывания (приходит примерно каждые 54 мс), запоминаются содержимые регистров, по номеру источника прерывания в таблице векторов определяется смещение, запоминается адрес 2 байта в IP и 2 байта в CS. Далее выполняется прерывание по сохранённому адресу и далее восстанавливается информация прерванного процесса и управление возвращается прерванной программе.

2) Какого типа прерывания использовались в программе?

Аппаратное прерывание 1Ch, int 10h, int 21h.