

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. 8383

Бабенко Н.С.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2020

Цель работы.

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

В лабораторной работе предлагается построить обработчик прерываний сигналов таймера. Эти сигналы генерируются аппаратурой через определённые интервалы времени и, при возникновении такого сигнала, возникает прерывание с определённым значением вектора. Таким образом, управление будет передано функции, чья точка входа записана в соответствующий вектор прерывания.

Сведения о программе.

Функции, используемые в программе:

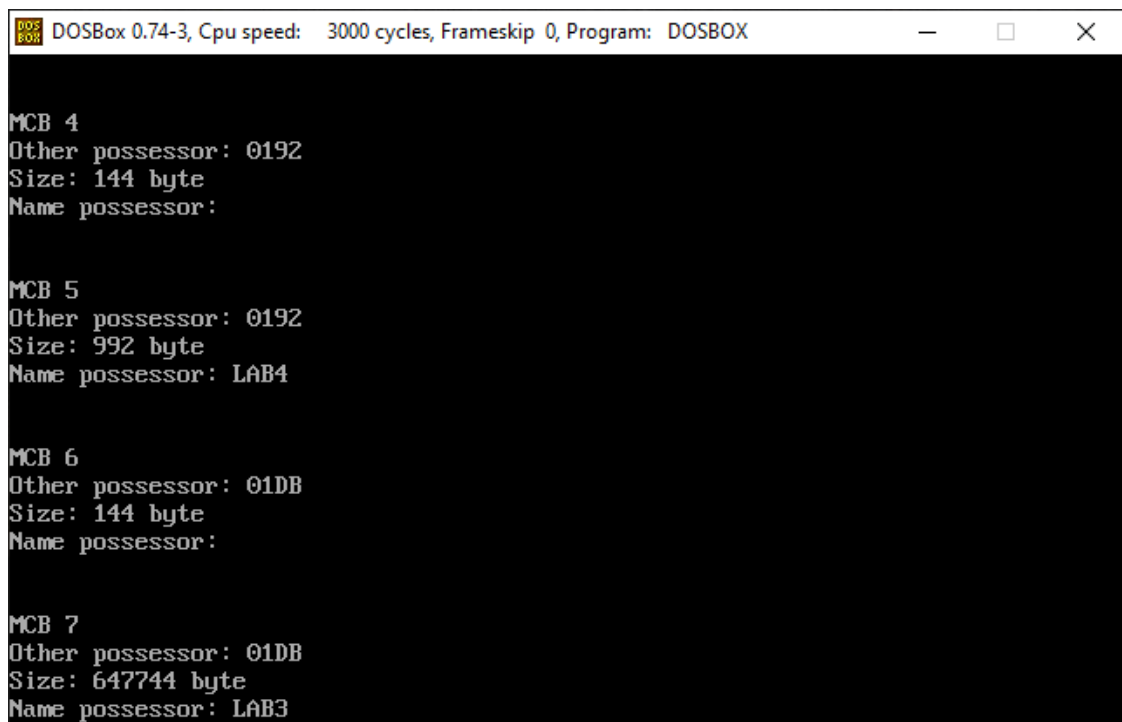
- 1) WRITE – выводит строку, адрес начала которой записан в DX.
- 2) PUTSYMB – выводит символ, записанный в AL в текущее положение курсора, которое характеризуется координатами, записанными в DH и DL.
- 3) PUTSYMB2 – выводит символ, записанный в AL в текущее положение курсора, которое характеризуется координатами, записанными в DH и DL, 2 раза.
- 4) GETPOS – получает текущее положение курсора, которое возвращается в регистрах DH и DL.
- 5) SETPOS – устанавливает текущее положение курсора по регистрам DH и DL.
- 6) INTCOUNT – собственный обработчик прерывания, который выводит общее количество прерываний.

Последовательность действий, выполняемых программой:

- 1) Проверка установки пользовательского прерывания.
- 2) Если оно не установлено, то в память резидентно загружается обработчик этого прерывания.
- 3) Если оно установлено, то происходит проверка ключа удаления.
- 4) Если в командной строке был ключ «/un», то происходит выгрузка резидента из памяти.

Ход выполнения.

- 1) Был запущен DOSBox.
- 2) В файле LAB4.asm была написана требуемая программа.
- 3) Из файла LAB4.asm был получен исполняемый модуль LAB4.exe.
- 4) Была запущена программа LAB4.exe. Для проверки нахождения программы в памяти была запущена программа LAB3.com из предыдущей лабораторной работы. На скриншоте представлен результат её работы и действия обработчика прерываний:



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX

MCB 4
Other possessor: 0192
Size: 144 byte
Name possessor:

MCB 5
Other possessor: 0192
Size: 992 byte
Name possessor: LAB4

MCB 6
Other possessor: 01DB
Size: 144 byte
Name possessor:

MCB 7
Other possessor: 01DB
Size: 647744 byte
Name possessor: LAB3
```

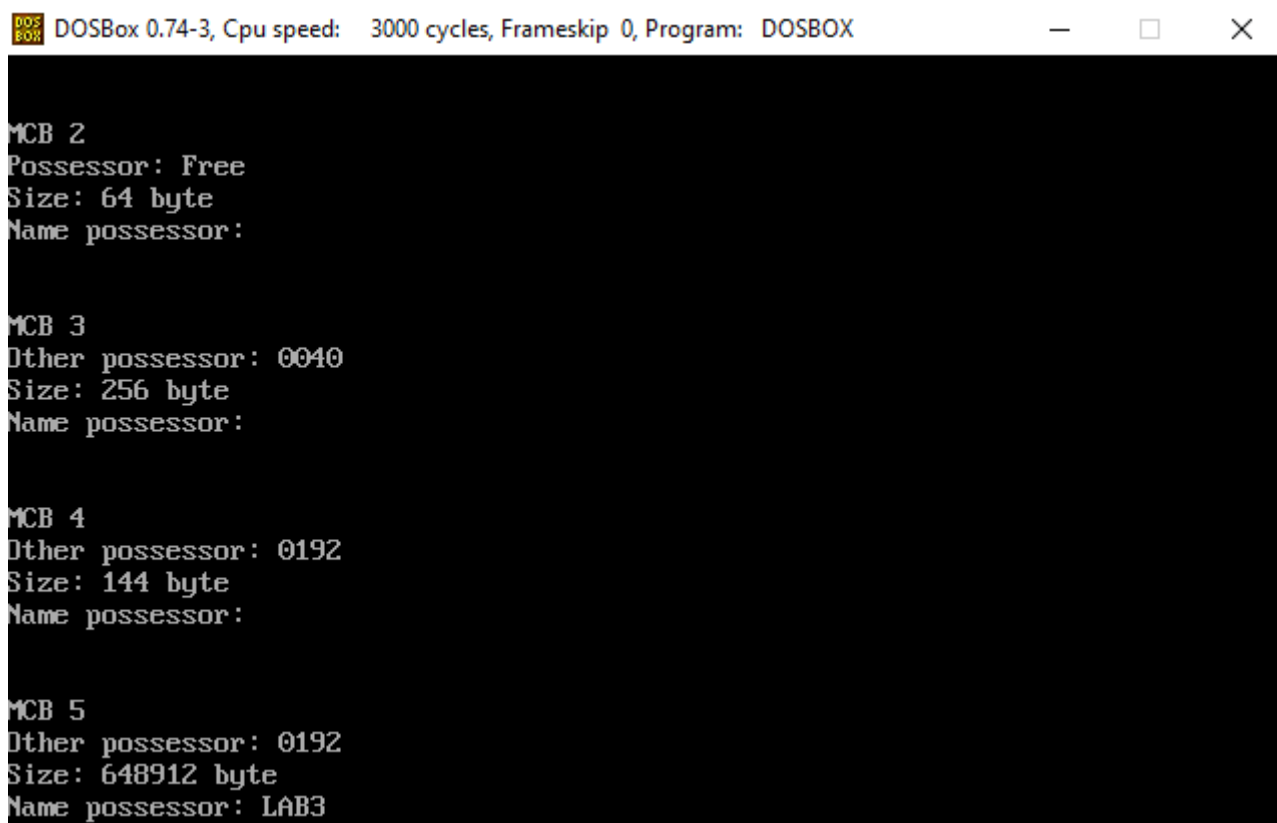
5) Снова была запущена программа lab4.exe. Результат её работы представлен на скриншоте:

```
C:\>lab4.exe  
Resident loaded into memory!
```

6) Теперь программа LAB4.exe была запущена с ключом удаления с помощью команды «LAB4.exe /un».

```
C:\>LAB4.EXE /un  
Resident removed from memory!
```

Для проверки нахождения программы в памяти была запущена программа LAB3.com из прошлой лабораторной работы. На скриншоте представлен результат её работы:



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX  
  
MCB 2  
Possessor: Free  
Size: 64 byte  
Name possessor:  
  
MCB 3  
Other possessor: 0040  
Size: 256 byte  
Name possessor:  
  
MCB 4  
Other possessor: 0192  
Size: 144 byte  
Name possessor:  
  
MCB 5  
Other possessor: 0192  
Size: 648912 byte  
Name possessor: LAB3
```

Контрольные вопросы.

- 1) Как реализован механизм прерывания от часов?
 - происходит увеличение счётчика системного таймера;
 - сохраняется контекст прерванного процесса;

- если они разрешены, то происходит вызов обработчика прерываний, адрес которого расположен в таблице векторов прерываний;
- происходит проверка на запрещение в данный момент прерываний этого типа, если они запрещены, то продолжается дальше выполнение текущей последовательности команд;
- временно запрещаются прерывания данного типа;
- восстанавливается контекст, разрешаются прерывания данного типа, возврат к выполнению текущей последовательности команд в то место, в котором был получен запрос прерывания.
- происходит сама обработка прерывания;

2) Какого типа прерывания использовались в работе?

1Ch	Аппаратное (асинхронное) прерывание. (Пользовательское прерывание по таймеру)
10h	Программное прерывание (прерывание BIOS)
21h	Программное прерывание (прерывание DOS)

Выводы.

В ходе выполнения данной лабораторной работы был исследован механизм работы прерываний и устройство резидентных программ, был реализован собственный обработчик прерываний от системного таймера, который резидентно загружается в память.