

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

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

Студентка гр. 0381

\_\_\_\_\_

Короткина Е.А.

Преподаватель

\_\_\_\_\_

Ефремов М.А.

Санкт-Петербург

2022

### **Цель работы.**

Построить обработчик прерываний сигналов таймера.

### **Постановка задачи.**

1. 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 осуществляет 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 функциям 4Ch прерывания int 21h.

Программа должна содержать код устанавливаемого прерывания в виде удаленной процедуры. Этот код должен выполнять следующие функции:

1) Сохранить 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

2. Запустить отлаженную программу и убедиться, что она работает. Проверить размещение прерывания в памяти с помощью программы, написанной в ЛР3.

3. Запустить программу еще раз и убедиться, что программа определяет установленный обработчик прерываний.

4. Запустить программу с ключом выгрузки и убедиться, что резидентный обработчик прерываний выгружен, а память, занятая резидентом, освобождена.

5. Ответить на контрольные вопросы

### **Выполнение работы.**

Для вывода сообщений написана процедура WRITEMESSAGE.

Для вывода строки из обработчика прерывания написана процедура outputBP. Для получения позиции курсора и установки его позиции написаны процедуры setCurs и getCurs соответственно.

Написан обработчик прерывания 1Ch. В обработчике организован стек на 512 слов. В слове CUSTOM хранится уникальный идентификатор обработчика, который позволяет отличить его от стандартного. В слова KEEP\_IP и KEEP\_CS будут записаны значения смещения и адреса сегмента стандартного обработчика прерываний при установке пользовательского обработчика. В строке INT\_NUM\_STRING хранится строка, в которой записано, сколько раз был вызван обработчик прерываний.

Написана процедура UPDATE\_COUNTER, которая обновляет количество вызовов прерывания в строке INT\_NUM\_STRING.

Пользовательское прерывание устанавливается в процедуре SET\_INTERRUPTION. В данной процедуре сохраняется адрес сегмента и смещение стандартного прерывания, чтобы его можно было восстановить, после чего устанавливает резидентную функцию на его место.

Для того, чтобы определить, установлен ли пользовательский обработчик прерываний, написана процедура CHECK\_CUSTOM. Эта функция проверяет значение слова CUSTOM и в зависимости от его значения определяет,

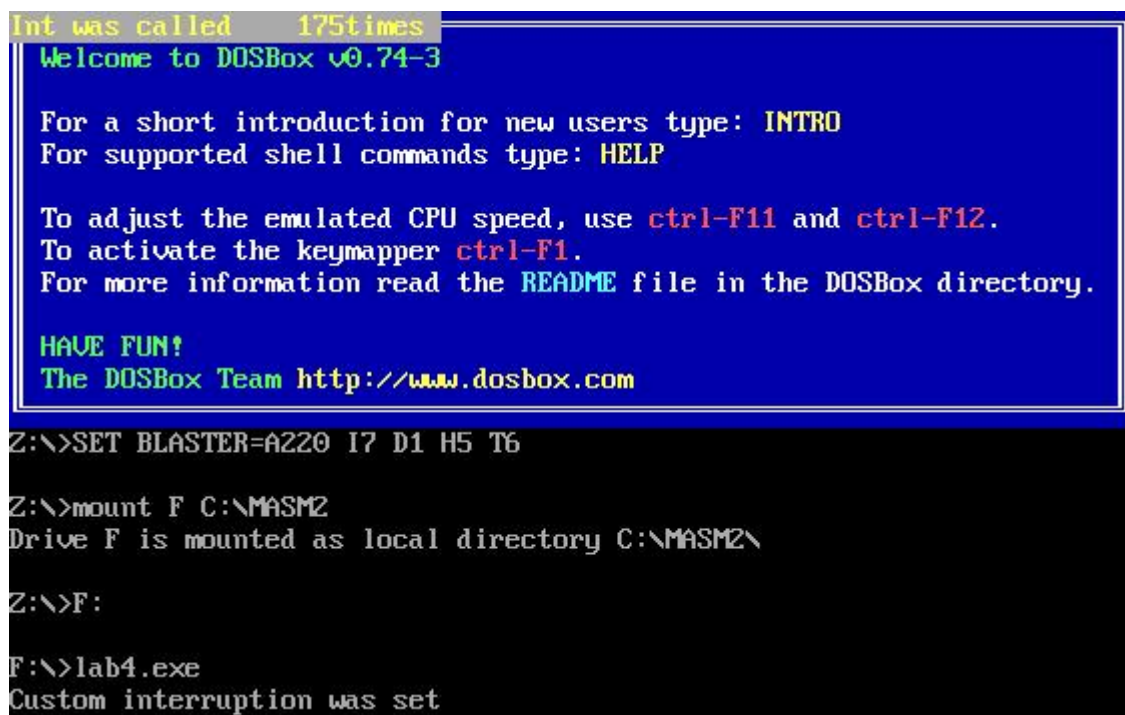
установлено пользовательское прерывание или нет. Если установлено, то в AL заносится значение 1, иначе - значение 0.

Для того, чтобы вернуть стандартное прерывание, написана процедура `DISABLE_CUSTOM`. Эта процедура будет вызвана, если программа запущена с параметром `/un`. Наличие этого параметра определяется процедурой `GET_COMMANDLINE_TAIL` на основе значения, лежащего со смещением 81h в PSP.

При запуске программа вызывает процедуры `CHECK_CUSTOM` и `CHECK_COMMANDLINE_TAIL`. Если пользовательский обработчик прерывания уже установлен, выводится сообщение «Custom interruption is set»; если обработчик не установлен, вызывается устанавливающая его процедура `SET_INTERRUPTION` и выводится сообщение «Custom interruption was set».

Если обработчик был установлен, то проверяется наличие параметра `/un`. Если он есть, вызывается процедура `DISABLE_CUSTOM`, после чего выводится сообщение «Custom interruption is no longer set».

Результат запусков программы см. Рис. 1-5.



```
Int was called 175times
Welcome to DOSBox v0.74-3

For a short introduction for new users type: INTRO
For supported shell commands type: HELP

To adjust the emulated CPU speed, use ctrl-F11 and ctrl-F12.
To activate the keymapper ctrl-F1.
For more information read the README file in the DOSBox directory.

HAVE FUN!
The DOSBox Team http://www.dosbox.com

Z:\>SET BLASTER=A220 I7 D1 H5 T6

Z:\>mount F C:\MASM2
Drive F is mounted as local directory C:\MASM2\

Z:\>F:

F:\>lab4.exe
Custom interruption was set
```

Рис.1. Запуск программы. Обработчик установлен.

```

Int was called 306times
Z:\>F:

F:\>lab4.exe
Custom interruption was set

F:\>lab3.com
Available memory size: 647216 byte
Extended memory size: 245760 byte
MCB || Type: 4Dh | Address: 016Fh | Size: 16      bytes | Owner: MS DOS
      | Other:
MCB || Type: 4Dh | Address: 0171h | Size: 64      bytes | Owner: Free
      | Other:
MCB || Type: 4Dh | Address: 0176h | Size: 256     bytes | Owner: 0040h
      | Other:
MCB || Type: 4Dh | Address: 0187h | Size: 144     bytes | Owner: 0192h
      | Other:
MCB || Type: 4Dh | Address: 0191h | Size: 1520    bytes | Owner: 0192h
      | Other: LAB4
MCB || Type: 4Dh | Address: 01F1h | Size: 1440    bytes | Owner: 01FCh
      | Other:
MCB || Type: 5Ah | Address: 01FBh | Size: 647216 bytes | Owner: 01FCh
      | Other: LAB3

```

Рис.2. Запуск модуля из ЛР3. Прерыванию выделена память в 5 строке (в поле Other - LAB4)

```

Custom interruption was set

F:\>lab3.com
Available memory size: 647216 byte
Extended memory size: 245760 byte
MCB || Type: 4Dh | Address: 016Fh | Size: 16      bytes | Owner: MS DOS
      | Other:
MCB || Type: 4Dh | Address: 0171h | Size: 64      bytes | Owner: Free
      | Other:
MCB || Type: 4Dh | Address: 0176h | Size: 256     bytes | Owner: 0040h
      | Other:
MCB || Type: 4Dh | Address: 0187h | Size: 144     bytes | Owner: 0192h
      | Other:
MCB || Type: 4Dh | Address: 0191h | Size: 1520    bytes | Owner: 0192h
      | Other: LAB4
MCB || Type: 4Dh | Address: 01F1h | Size: 1440    bytes | Owner: 01FCh
      | Other:
MCB || Type: 5Ah | Address: 01FBh | Size: 647216 bytes | Owner: 01FCh
      | Other: LAB3

F:\>lab4.exe
Custom interruption is set

```

Рис.3. Повторный запуск программы. Пользовательский обработчик был определен.

```

Available memory size: 647216 byte
Extended memory size: 245760 byte
MCB || Type: 4Dh | Address: 016Fh | Size: 16      bytes | Owner: MS DOS
      | Other:
MCB || Type: 4Dh | Address: 0171h | Size: 64      bytes | Owner: Free
      | Other:
MCB || Type: 4Dh | Address: 0176h | Size: 256     bytes | Owner: 0040h
      | Other:
MCB || Type: 4Dh | Address: 0187h | Size: 144     bytes | Owner: 0192h
      | Other:
MCB || Type: 4Dh | Address: 0191h | Size: 1520    bytes | Owner: 0192h
      | Other: LAB4
MCB || Type: 4Dh | Address: 01F1h | Size: 1440    bytes | Owner: 01FCh
      | Other:
MCB || Type: 5Ah | Address: 01FBh | Size: 647216 bytes | Owner: 01FCh
      | Other: LAB3

F:\>lab4.exe
Custom interruption is set

F:\>lab4.exe /un
Custom interruption is set
Custom interruption is no longer set

```

Рис.4. Запуск с параметром /un. Пользовательский обработчик был выгружен.

```

MCB || Type: 5Ah | Address: 01FBh | Size: 647216 bytes | Owner: 01FCh
      | Other: LAB3

F:\>lab4.exe
Custom interruption is set

F:\>lab4.exe /un
Custom interruption is set
Custom interruption is no longer set

F:\>lab3.com
Available memory size: 648912 byte
Extended memory size: 245760 byte
MCB || Type: 4Dh | Address: 016Fh | Size: 16      bytes | Owner: MS DOS
      | Other:
MCB || Type: 4Dh | Address: 0171h | Size: 64      bytes | Owner: Free
      | Other:
MCB || Type: 4Dh | Address: 0176h | Size: 256     bytes | Owner: 0040h
      | Other:
MCB || Type: 4Dh | Address: 0187h | Size: 144     bytes | Owner: 0192h
      | Other:
MCB || Type: 5Ah | Address: 0191h | Size: 648912 bytes | Owner: 0192h
      | Other: LAB3

```

Рис.5. Запуск модуля из ЛР3. Память была очищена.

Ответы на контрольные вопросы:

1. Как реализован механизм прерывания от часов?

По каждому тикку аппаратных часов (каждые 55 миллисекунд или примерно 18.2 раз в секунду) аппаратными средствами подается сигнал и вызывается прерывание 1Ch. Изначально в нем стоит «заглушка» в виде IRET, но туда может быть загружено пользовательское прерывание.

2. Какого типа прерывания использованы в работе?

В работе использованы программные прерывания 10h, 21h и аппаратное прерывание 1Ch.

**Вывод.**

Был построен обработчик прерываний сигналов таймера.