

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.

Студент гр. 8381

Перелыгин Д.С.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2020

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.

В лабораторной работе №4 предлагается построить обработчик прерываний сигналов таймера. Эти сигналы генерируются аппаратурой через определенные интервалы времени и, при возникновении такого сигнала, возникает прерывание с определенным значением вектора. Таким образом, управление будет передано функции, чья точка входа записана в соответствующий вектор прерывания.

Функции и структуры данных

1. Процедура PRINT — осуществляет вывод текста на экран.
2. Процедура ROUT — пользовательский обработчик прерываний. Выводит на экран информацию о количестве срабатываний.
3. Процедура outputAL — выводит символ на экран.
4. Процедура setInt — устанавливает пользовательский обработчик прерываний.
5. Процедура restoreInt — восстанавливает стандартный обработчик прерываний и очищает память, занятую программой-обработчиком.
6. Процедура Main — дальнейший код выполняет главную задачу программы: выводит на экран необходимую информацию.

Последовательность выполняемых действий.

1. Проверка установленного в данный момент обработчика прерываний. Если он пользовательский, то по заданному смещению в нем содержится определенная сигнатура.

2. Если сигнатура не найдена, то устанавливается пользовательский обработчик прерываний.
3. Если найдена, то выводится соответствующее сообщение в консоль. Далее проверяется хвост командной строки на наличие команды удаления пользовательского обработчика прерываний. Если команда присутствует, то восстанавливается стандартный обработчик, а память, выделенная под резидентную программу, освобождается.

Результаты, полученные при первом запуске программы в эмуляторе DOSBox. Программа выводит информацию о том, что сейчас действует стандартный обработчик прерываний и устанавливает резидента. Также вызвана программа lab3 для контроля блоков памяти.

```
C:\>LR4.EXE
Resident is not uploaded

C:\>LAB3_1.COM

Size of accessed memory: 648224 byte
Size of extended memory: 245760 byte
MCB:01 Address: 016F PSP address: 0008 Size: 16 SD/SC:
MCB:02 Address: 0171 PSP address: 0000 Size: 64 SD/SC:
MCB:03 Address: 0176 PSP address: 0040 Size: 256 SD/SC:
MCB:04 Address: 0187 PSP address: 0192 Size: 144 SD/SC:
MCB:05 Address: 0191 PSP address: 0192 Size: 512 SD/SC: LR4
MCB:06 Address: 01B2 PSP address: 01BD Size: 144 SD/SC:
MCB:07 Address: 01BC PSP address: 01BD Size: 648224 SD/SC: LAB3_1
C:\>
```

Второй запуск, без команды /un. Выводится информация о том, что резидент уже установлен, ничего не происходит. Из вывода программы lab3 видно, что память не изменилась.

```

C:\>LR4.EXE
Resident is uploaded
02
C:\>LAB3_1.COM

Size of accessed memory: 648224 byte
Size of extended memory: 245760 byte
MCB:01 Address: 016F PSP address: 0008 Size: 16 SD/SC:
MCB:02 Address: 0171 PSP address: 0000 Size: 64 SD/SC:
MCB:03 Address: 0176 PSP address: 0040 Size: 256 SD/SC:
MCB:04 Address: 0187 PSP address: 0192 Size: 144 SD/SC:
MCB:05 Address: 0191 PSP address: 0192 Size: 512 SD/SC: LR4
MCB:06 Address: 01B2 PSP address: 01BD Size: 144 SD/SC:
35 MCB:07 Address: 01BC PSP address: 01BD Size: 648224 SD/SC: LAB3_1
C:\>_

```

Третий запуск, с командой /un. Программа сообщает о том, что резидент существует, после чего уничтожает его. Память также освобождается.

```

C:\>LR4.EXE /un
Resident is uploaded
Resident was unloaded

C:\>LAB3_1.COM

Size of accessed memory: 648912 byte
Size of extended memory: 245760 byte
MCB:01 Address: 016F PSP address: 0008 Size: 16 SD/SC:
MCB:02 Address: 0171 PSP address: 0000 Size: 64 SD/SC:
MCB:03 Address: 0176 PSP address: 0040 Size: 256 SD/SC:
MCB:04 Address: 0187 PSP address: 0192 Size: 144 SD/SC:
MCB:05 Address: 0191 PSP address: 0192 Size: 648912 SD/SC: LAB3_1
C:\>_

```

Контрольные вопросы.

1. Как реализован механизм прерывания от часов?

Каждый определенный промежуток времени сохраняется состояние регистров, по таблице векторов прерываний определяется функция-обработчик прерывания, ее адрес записывается в регистры CS и IP. После обработки прерывания сохраненные регистры восстанавливаются, управление передается прерванному процессу.

2. Какого типа прерывания использовались в работе?

- Аппаратные прерывания – асинхронное прерывание от таймера;
- Прерывания функций DOS(21h);
- Прерывания функций BIOS(10h).

Выводы.

В ходе работы были изучены механизмы прерывания, основы реализации резидентных программ.