

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

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

Студент гр. 9381

Авдеев И.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2021

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.

Ход работы.

Был написан текст исходного EXE модуля lab4.asm. При помощи транслятора MASM.EXE и компоновщика LINK.EXE был скомпилирован EXE модуль lab4.com, после чего была осуществлена отладка. Программа выполняет следующие функции:

- Проверяет, установлено ли пользовательское прерывание с вектором 1Ch.
- Устанавливает резидентную функцию для обработки прерывания и настраивает вектор прерываний, если прерывание не установлено, и осуществляется выход о функции 4Ch прерывания int 21h.
- Если прерывание установлено, то выводится соответствующее сообщение и осуществляется выход по функции 4Ch прерывания int 21h.
- Выгрузка прерывания о 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.

Функции и структуры данных.

Разработанная программа использует следующие функции и структуры данных:

Названия процедур	Описание
INT_COUNT_FUNC	Выводит количество вызванных прерываний

CHECK_INTSET	Проверяет наличие уже установленного прерывания
CHECK_UN	Проверяет наличие параметра /un и возвращает, значение, по которому в функции MAIN происходит загрузка/выгрузка прерываний
REDEF_INT	Устанавливает новые обработчики прерывания (переопределяет) при помощи функции 25h прерывания int 21h
RESTORE_INT	Восстанавливает сохраненные обработчики прерываний и выгружает резидентную функцию
WRITE	Выводит строку в консоль
MAIN	Главная процедура

Тестирование.

1)Проверялась установка нового резидентного обработчика прерывания 1Ch. Для проверки того, что обработчик прерываний остался резидентным в памяти, использовалась программа LAB3_1.COM. Были получены следующие результаты:

```

DOS BOX DOSBox 0.74-3, Cpu speed: 3000 cycles, Fram...
Z:\>I:
Drive I does not exist!
You must mount it first. Type inCount of interruptions: 0168information.

Z:\>mount I I:\Audeev_Ilya\OS\lr_4
Drive I is mounted as local directory I:\Audeev_Ilya\OS\lr_4\

Z:\>I:

I:\>lab4.exe
INTERRUPTION IS LOADING AT THIS MOMENT

I:\>lab3_1.com
Accesible memory: 648112 bytes
Extended memory: 15360 kilobytes

MCB type: 4D PSP address: 0008 Size: 16 bytes SC/SD:
MCB type: 4D PSP address: 0000 Size: 64 bytes SC/SD:
MCB type: 4D PSP address: 0040 Size: 256 bytes SC/SD:
MCB type: 4D PSP address: 0192 Size: 144 bytes SC/SD:
MCB type: 4D PSP address: 0192 Size: 624 bytes SC/SD: LAB4
MCB type: 4D PSP address: 01C4 Size: 144 bytes SC/SD:
MCB type: 5A PSP address: 01C4 Size: 648112 bytes SC/SD: LAB3_1

I:\>

```

2)Проверялось определение программой установленного обработчика прерывания 1Ch. Были получены следующие результаты:

```

DOS BOX DOSBox 0.74-3, Cpu speed: 3000 cycles, Fram...
Z:\>mount I I:\Audeev_Ilya\OS\lr_4
Drive I is mounted as local direCount of interruptions: 0408\

Z:\>I:

I:\>lab4.exe
INTERRUPTION IS LOADING AT THIS MOMENT

I:\>lab3_1.com
Accesible memory: 648112 bytes
Extended memory: 15360 kilobytes

MCB type: 4D PSP address: 0008 Size: 16 bytes SC/SD:
MCB type: 4D PSP address: 0000 Size: 64 bytes SC/SD:
MCB type: 4D PSP address: 0040 Size: 256 bytes SC/SD:
MCB type: 4D PSP address: 0192 Size: 144 bytes SC/SD:
MCB type: 4D PSP address: 0192 Size: 624 bytes SC/SD: LAB4
MCB type: 4D PSP address: 01C4 Size: 144 bytes SC/SD:
MCB type: 5A PSP address: 01C4 Size: 648112 bytes SC/SD: LAB3_1

I:\>lab4.exe
INTERRUPTION IS ALREADY LOAD

I:\>_

```

3) Проверялась очистка памяти, занимаемой резидентным обработчиком

прерываний. Для этого была запущена программа с ключом загрузки “un”. Для проверки освобождения памяти, занятой резидентом, снова был запущен исполняемый .com файл ЛР3. Были получены следующие результаты:

```
I:\>lab4.exe /un
INTERRUPTION RESTORED

I:\>lab3_1.com
Accessible memory: 648912 bytes
Extended memory: 15360 kilobytes

MCB type: 4D    PSP address: 0008    Size: 16 bytes    SC/SD:
MCB type: 4D    PSP address: 0000    Size: 64 bytes    SC/SD:
MCB type: 4D    PSP address: 0040    Size: 256 bytes   SC/SD:
MCB type: 4D    PSP address: 0192    Size: 144 bytes   SC/SD:
MCB type: 5A    PSP address: 0192    Size: 648912 bytes SC/SD: LAB3_1

I:\>_
```

Ответы на контрольные вопросы.

1. Как реализован механизм прерывания от часов?
 - При возникновении сигнала аппаратного прерывания от часов, происходит первичное аппаратное распознавание типа прерывания. Если прерывания данного типа в настоящий момент запрещены, то процессор продолжает естественный ход выполнения команд, иначе происходит вызов обработчика прерывания 1Ch, адрес которой находится в таблице векторов прерываний;
 - Автоматически сохраняется некоторая часть контекста прерванного процесса, который позволит ядру возобновить его исполнение после обработки прерывания;
 - Временно запрещаются прерывания данного типа, чтобы не было очереди потоков одной и той же процедуры;
 - После обработки прерываний ядром ОС, контекст прерванного процесса восстанавливается, и его работа возобновляется с прерванного места; также снимается блокировка прерываний данного типа.
2. Какого типа прерывания использовались в работе?

Использовались следующие типы прерываний:

 - Аппаратные (прерывание от часов – 1Ch);
 - Программные (прерывания, вызываемые при помощи команды int – 21h)

Вывод.

В ходе выполнения работы была написана программа обработчика прерывания от сигналов таймера и освоена техника установки резидентной программы в память и её выгрузка из памяти. Помимо этого была более подробно изучена обработка стандартных прерываний.