**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. 7381 |  | Судакова П.С. |
| Преподаватель |  | Ефремов М. А. |

Санкт-Петербург

2019

# **Цель работы.**

В лабораторной работе 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.

# **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 о 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 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.

**Шаг 2**. Далее необходимо запустить отлаженную программу и убедиться, что резидентный обработчик прерывания 1Ch установлен. Работа прерывания должна отображаться на экране, а также необходимо проверить размещение прерывания в памяти. Для этого нужно запустить программу ЛР 3, которая отображает карту памяти в виде с писка блоков MCB.

**Шаг 3**. Затем необходимо запустить отлаженную программу еще раз и убедиться, что программа определяет установленный обработчик прерываний.

**Шаг 4**. Далее нужно запустить отлаженную программу с ключом выгрузки и убедиться, 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.

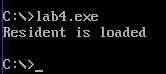
**Сведения о функциях, используемых в программе**

**Сведения о функциях и структурах данных, используемых в программе**

|  |  |  |
| --- | --- | --- |
| *Название поля* | *Тип* | *Назначение* |
| COUNT\_STR | db | Хранение строки: “THIS IS MY CUR COUNT: 0000” |
| NOT\_DOWNLOAD\_STR | db | Хранение строки: “Resident didnt download” |
| RESTORED\_STR | db | Хранение строки: “Resident was restored” |
| ALREADY\_STR | db | Хранение строки: “Resident is already loaded” |
| LOADED\_STR | db | Хранение строки: “Resident is loaded” |
| INIT\_FUNC | PROC | Устанавливает резидентную функцию для обработки прерываний |
| SETTING\_FUNC | PROC | Проверяет установлено ли прерывание |
| CHECKING\_END | PROC | Осуществляет проверку на наличие префикса /un |
| LOAD\_FUNC | PROC | Загружает прерывание |
| UNLOAD\_FUNC | PROC | Выгружает прерывание |
| PRINT | PROC | Вывод на экран |
| MAIN\_FUNC | PROC | Функция начала программы |

# **Работа программы.**

Загрузим резидент в память

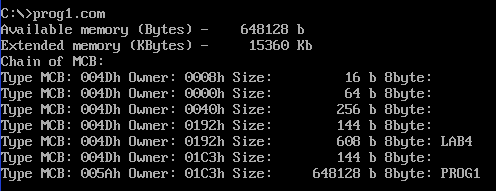


prog

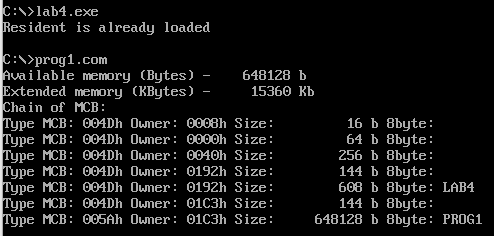
Работа счетчика



Размещение прерывания в памяти



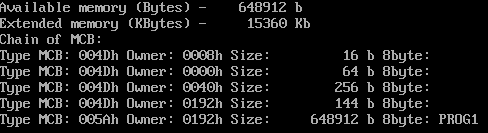
Загрузим резидент еще раз и получим следующее предупреждение и затем выведем размещение в памяти



Выгрузим резидент из памяти



После этого получим следующее содержимое памяти



# **Контрольные вопросы.**

1. Как реализован механизм прерывания от часов?

**Ответ:** При каждом такте таймера (часов) происходит следующее: сначала сохраняется состояние регистров, затем определяется источник прерывания (по номеру источника прерывания определяется смещение в таблице векторов прерываний), далее эти данные сохраняются, т. е. - первые два байта помещаются в IP, второе два байта в CS, далее запускается обработчик прерывания по адресу CS:IP и обрабатывается прерывание, в конце происходит возврат управления прерванной программе

1. Какого типа прерывания использовались в работе?

**Ответ:** Были использованы пользовательские прерывания, такие как int 10h и int 21h и аппаратные прерывания (1Ch).

# **Выводы.**

В результате выполнения данной лабораторной работы были исследованы организация и управление прерываниями. Разработана программа, в которой реализован обработчик прерываний сигналов таймера.