**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МОЭВМ**

Лабораторная работа №4

**по дисциплине «Операционные системы»**

Тема: «**Обработка стандартных прерываний»**

|  |  |  |
| --- | --- | --- |
| Студентка гр. 6383 |  | Гомонова А.А. |
| Преподаватель |  | Губкин А.Ф. |

Санкт-Петербург

2018

1. **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 прерывания 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 определённым значением вектора. Таким образом, управление будет передано функции, чья точка входа записана в соответствующий вектор прерывания.

1. **Необходимые сведения для составления программы:**

Резидентные обработчики прерываний - это программные модули, которые вызываются при возникновении прерываний определенного типа (сигнал таймера, нажатие клавиши и т.д.), которым соответствуют определенные вектора прерывания. Когда вызывается прерывание, процессор переключается на выполнение кода обработчика, а затем возвращается на выполнение прерванной программы. Адрес возврата в прерванную программу (CS:IP) запоминается в стеке вместе с регистром флагов. Затем в CS:IP загружается адрес точки входа программы обработки прерывания и начинает выполняться его код. Обработчик прерывания должен заканчиваться инструкцией IRET (возврат из прерывания).

Вектор прерывания имеет длину 4 байта. В первом хранится значение IP, во втором - CS. Младшие 1024 байта памяти содержат 256 векторов. Вектор для прерывания 0 начинается с ячейки 0000:0000, для прерывания 1 - с ячейки 0000:0004 и т.д.

1. **Постановка задачи и ход работы.**

Для выполнения лабораторной работы необходимо написать и отладить программный модуль типа .EXE, который выполняет следующие функции:

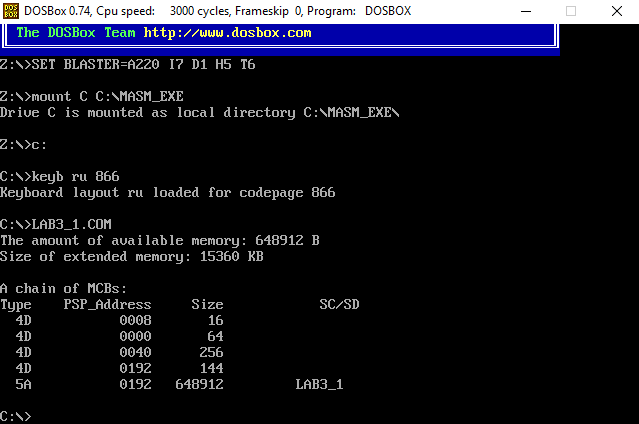
1. Проверяет, установлено ли пользовательское прерывание с вектором 1Ch.
2. Устанавливает резидентную функцию для обработки прерывания и настраивает вектор прерываний, если прерывание не установлено, и осуществляется выход по функции 4Ch прерывания int 21h.
3. Если прерывание установлено, то выводится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

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.

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

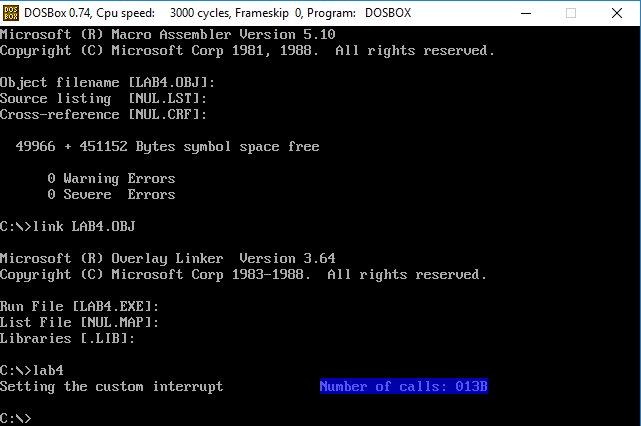
Далее нужно запустить отлаженную программу с ключом выгрузки и убедиться, что резидентный обработчик прерывания выгружен, то есть сообщения на экран не выводятся, а память, занятая резидентом освобождена. Для этого также следует запустить программу ЛР 3.

1. **Результаты работы программы**
2. Состояние памяти до загрузки резидента (используем модуль, разработанный в третьей лабораторной работе). (Рис. 1)



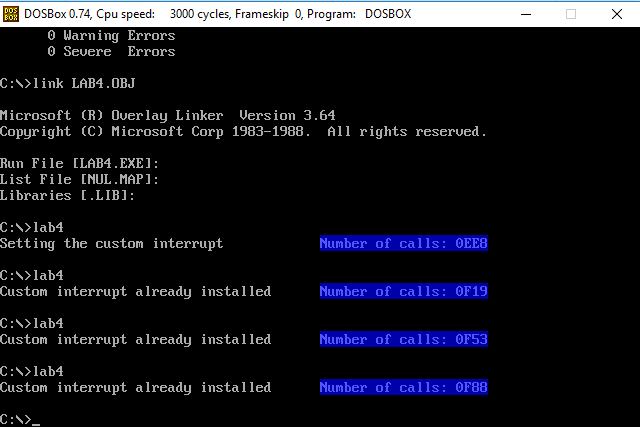
**Рисунок 1**

1. Загрузка резидента в память. (Рис. 2)



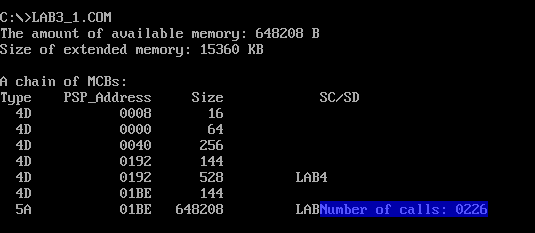
**Рисунок 2**

1. Попытка повторной загрузки резидента в память. (Рис. 3)

****

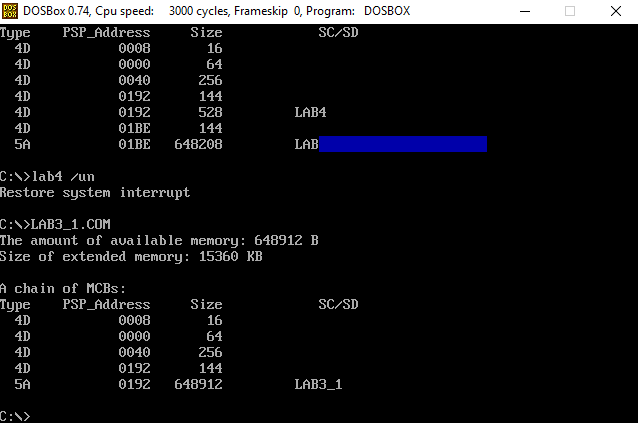
**Рисунок 3**

1. Состояние памяти при загрузке в неё резидента. (Рис. 4)

****

**Рисунок 4**

5) Запускаем отложенную программу с ключом /un, тем самым выгружаем резидент и смотрим состояние памяти после выгрузки резидента. (Рис. 5)



**Рисунок 5**

1. **Ответы на контрольные вопросы**
2. *Как реализован механизм прерывания от часов?*

Каждые 55 мс:

* 1. сохраняется состояние регистров
  2. определяется источник прерывания (по номеру источника прерывания определяется смещение в таблице векторов прерываний)
  3. первые два байта помещаются в IP, второе два байта в CS
  4. передаётся управление по адресу CS:IP (т.е. «запускается» обработчик прерывания)
  5. обработка прерывания
  6. возврат управления прерванной программе

1. *Какого типа прерывания использовались в работе?*

В работе использовались аппаратные прерывания (int 1Ch) и пользовательские прерывания (int 21h, int 10h).

1. **Вывод**

В ходе данной лабораторной работы был построен собственный обработчик прерывания для аппаратного прерывания 1Ch, происходящего по сигналу системного таймера.