**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. 6381 |  | Токун М. С. |
| Преподаватель |  | Губкин А. Ф. |

Санкт-Петербург

2018

**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 и выполняет соответствующие действия.

В лабораторной работе номер 4 предлагается построить обработчик прерываний сигналов таймера. Эти сигналы генерируются аппаратурой через определённые 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.

**Ход работы.**

## **Описание данных**

**SIGNATURE** – подпись пользовательского прерывания.

**UNLOAD –** флаг для выгрузки прерывания.

**KEEP\_IP –** для сохранения регистра ip.

**KEEP\_CS –** для сохранения регистра cs.

**KEEP\_ES –** для сохранения регистра es.

**KEEP\_SP –** для сохранения регистра sp.

**KEEP\_SS –** для сохранения регистра ss.

**OUT\_STRING –** строка для вывода сообщения о количестве прерываний.

**ONES –** разряд единиц.

**DOZENS –** разряд десятков.

**HUNDREDS –** разряд сотен.

**THOUSANDS –** разряд тысяч.

**LOADED –** сообщение о загрузке прерывания.

**UNLOADED –** сообщение о выгрузке прерывния.

**LOADED\_BEFORE –** сообщение о том что прерывание уже установлено.

## **Описание функций**

**MY\_INT –** пользовательский обработчик прерываний

**CALCULATE\_INT** – процедура подсчета количества вызванных прерываний.

**OUTPUT\_BP –** процедура для вывода строки на консоль в указанную позицию.

**CHECK –** проверяет установлено ли пользовательское прерывание и введен ли параметр /un.

**SET\_MY\_INT –** устанавливает пользовательское прерывание.

**SET\_DEFAULT\_INT –** восстанавливает исходное прерывание.

**WRITE –** процедура для вывода строки на консоль.

1. **Последовательность действий, выполняемых программой:**

Был написал и отлажен программный модуль типа ЕХЕ, который проверяет установлено ли пользовательское прерывание с вектором 1ch, если нет – то устанавливает резидентную функцию для обработки прерывания и настраивает вектор прерывания и осуществляется выход по функции 4ch прерывания int 21h его, после этого программа завершается. Если программа запущена с параметром /un, то пользовательское прерывание выгружается, и загружается исходное прерывание.

## **Результат выполнения**

1. Запуск программы

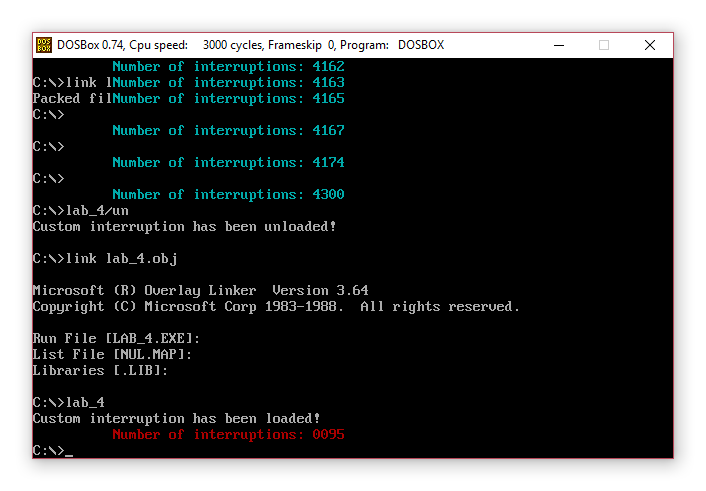


Рис.1 – Результат работы программы.

1. Результат работы вместе с программой из ЛР3.

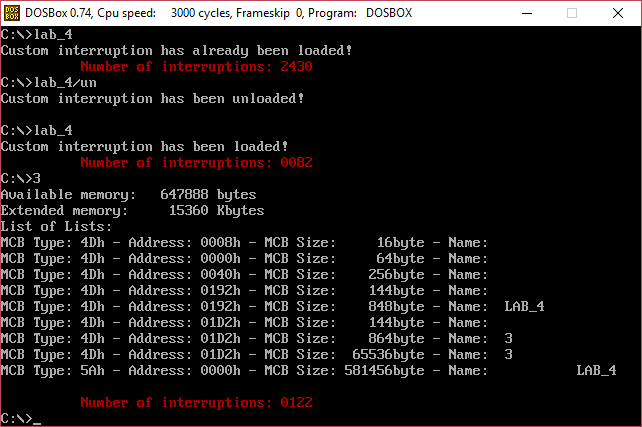


Рис.2 – Размещение прерывания в памяти

1. Запуск программы еще раз

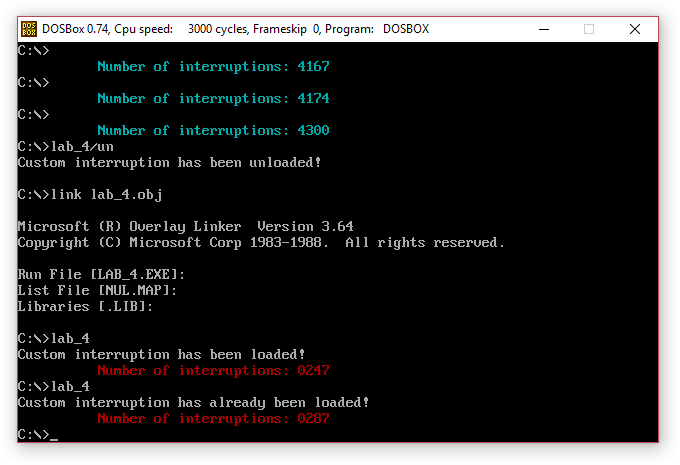


Рис.3 – Результат повторной установки прерывания.

1. Запуск с ключом выгрузки.

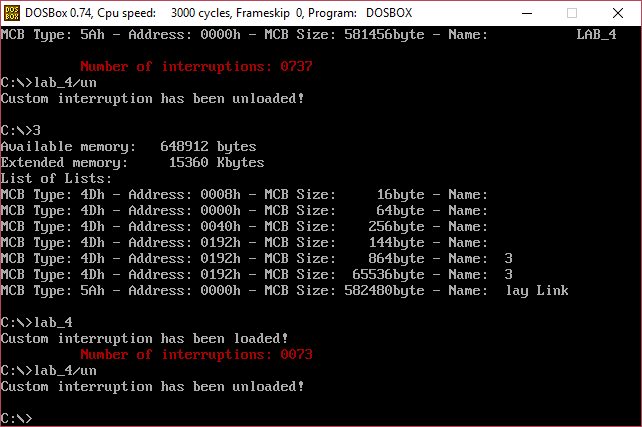


Рис.4 – Результат работы программы с ключом выгрузки.

**Вывод.**

В ходе данной лабораторной работы был создан обработчик прерываний сигналов таймера.

**Ответы на контрольные вопросы.**

1. *Как реализован механизм прерывания от часов?*

Механизм прерывания часов реализован следующим образом: Прерывание от часов (1сh) вызывается при каждом тике системного таймера (примерно 18 раз в секунду). Затем сохраняется состояние регистров, управление передается по смещению 70, затем выполняется код обработчика прерываний, далее восстанавливаются сохраненные регистры и выполняется возврат к управлению прерванной программой.

1. *Какого типа прерывания использовались в работе?*

В работе использованы аппаратные (1ch) и программные (10h, 21h) прерывания.