

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

Студентка гр. 0381

Степанова Е.М.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2022

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.

Задание.

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 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 по 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.

Программа должна содержать код устанавливаемого прерывания в виде удаленной процедуры. Этот код будет работать после установки при возникновении прерывания. Он должен выполнять следующие функции:

1. Сохраняет стек прерванной программы (регистры SS и SP) в рабочих переменных и восстановить при выходе.
2. Организовать свой стек.
3. Сохранить значения регистров в стеке при входе и восстановить их при выходе.
4. При выполнении тела процедуры накапливать общее суммарное число прерываний и выводить на экран. Для вывода на экран следует использовать прерывание int 10h, которое позволяет непосредственно выводить информацию на экран.
5. Функция прерывания должна содержать только переменные, которые она использует.

Шаг 2. Запустите отлаженную программу и убедитесь, что резидентный обработчик прерывания 1Ch установлен. Работа прерывания должна отображаться на экране, а также необходимо проверить размещение прерывания в памяти. Для этого запустите программу ЛР 3, которая отображает карту памяти в виде списка блоков МСВ. Полученные результаты поместите в отчет.

Шаг 3. Запустите отлаженную программу еще раз и убедитесь, что программа определяет установленный обработчик прерываний. Полученные результаты поместите в отчет.

Шаг 4. Запустите отлаженную программу с ключом выгрузки и убедитесь, 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. Полученные результаты поместите в отчет.

Выполнение работы.

Шаг 1. При запуске программы с помощью процедуры CHECK_FLAG проверяется наличие флага выгрузки /un. Для проверки используется посимвольное сравнение с образцом. Если флаг был установлен, процедура возвращает 1 в регистре al, иначе – 0.

Далее происходит проверка на наличие уже установленного пользовательского прерывания, для этого используется функция ISLOADED. Эта функция обращается во вектору прерывания и проверяет значение идентификатора. В пользовательском прерывании идентификатор имеет значение 3452h, а в стандартном его нет, поэтому, если при проверке значение совпадает, считается, что пользовательское прерывание установлено, и функция возвращает 1.

Далее, при помощи условных переходов определяется необходимое действие.

Если прерывание не загружено, и не установлен флаг выгрузки, выводится сообщение о том, что сейчас будет загружено прерывание и вызывается функция LOAD.

Если прерывание не загружено, и установлен флаг выгрузки, выводится сообщение о том, что прерывание не загружено, и программа завершается.

Если прерывание загружено, и флаг выгрузки не установлен, выводится сообщение о том, что прерывание уже загружено, и программа завершается.

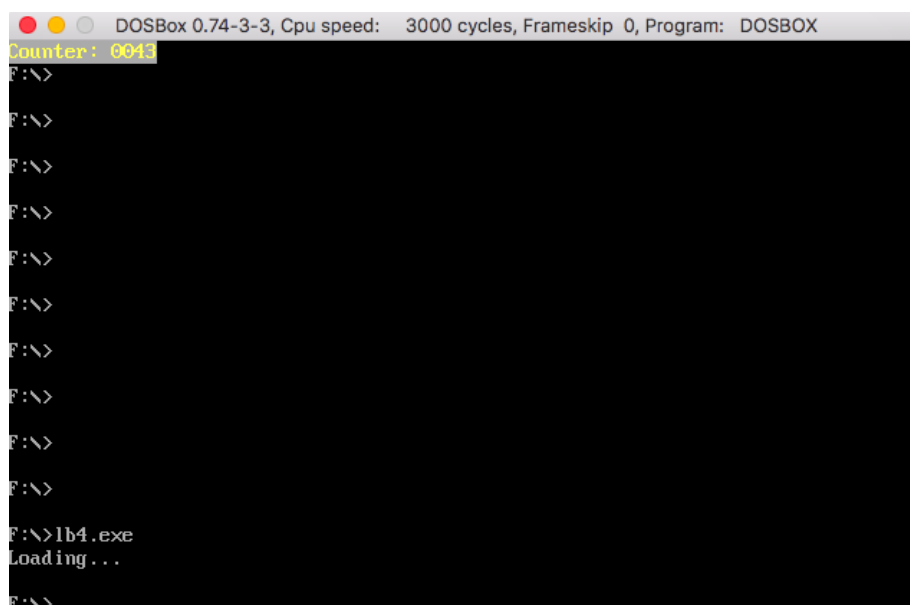
Если прерывание загружено, и установлен флаг выгрузки, выводится сообщение о том, что прерывание будет выгружено, и вызывается функция UNLOAD.

Функция LOAD сохраняет в теле прерывания стандартный вектор прерывания, устанавливает пользовательское прерывание с помощью функции 25h int 21h, и выходит из программы, оставляя её резидентной. Объем памяти для резидентной программы определяется по метке в конце пользовательского прерывания.

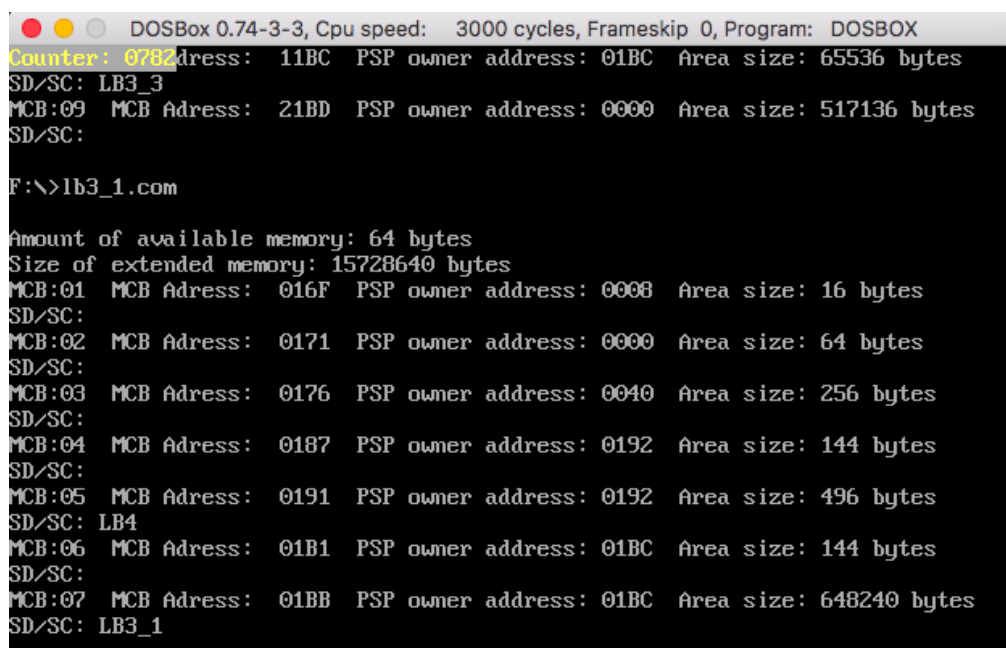
Функция UNLOAD обращается к сохраненным в теле прерывания переменным и восстанавливает по ним стандартный вектор прерывания. Затем, с помощью функции 49h int 21h она освобождает память, занимаемую прерыванием.

В теле прерывания хранятся адрес PSP владельца, адрес стандартного прерывания, идентификатор, переменные для сохранения регистров стека, строка для вывода на экран и стек прерывания. При выполнении прерывания сохраняются значения регистров SS, SP, AX. Далее настраивается стек прерывания, и в него сохраняются все регистры. Затем прерывание добавляет единицу к выводимому значению и устанавливает курсор в нужную позицию. Далее сообщение выводится на экран и восстанавливается предыдущая позиция курсора. В конце прерывания восстанавливаются значения регистров и сбрасывается контроллер прерываний.

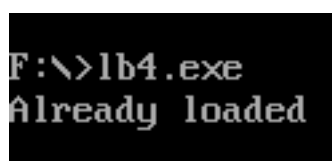
Шаг 2. Программа выводит счетчик в левом верхнем углу экрана.



Прерывание размещается в памяти по адресу 0192h. При запуске ЛР 3 счетчик на экране продолжает работать.



Шаг 3. При повторном запуске программы выводится сообщение о том, что обработчик уже установлен.



Шаг 4. При запуске с флагом выгрузки программа восстанавливает стандартное прерывание и очищает память.

```
F:\>lb4.exe /un
Unloading...

F:\>lb3_1.com


Amount of available memory: 64 bytes
Size of extended memory: 15728640 bytes
MCB:01 MCB Address: 016F PSP owner address: 0008 Area size: 16 bytes
SD/SC:
MCB:02 MCB Address: 0171 PSP owner address: 0000 Area size: 64 bytes
SD/SC:
MCB:03 MCB Address: 0176 PSP owner address: 0040 Area size: 256 bytes
SD/SC:
MCB:04 MCB Address: 0187 PSP owner address: 0192 Area size: 144 bytes
SD/SC:
MCB:05 MCB Address: 0191 PSP owner address: 0192 Area size: 648912 bytes
SD/SC: LB3_1
```

Выводы.

В ходе лабораторной работы была исследована обработка стандартных прерываний, а также построен обработчик прерываний сигналов таймера, которые генерируются аппаратурой через определённые интервалы времени. Программа загружает и выгружает резидентное прерывание, а также производится проверка флага выгрузки.

ВОПРОСЫ

1. Как реализован механизм прерывания от часов?

Принимается сигнал прерывания (приходит примерно каждые 54 мс), запоминаются содержимые регистров, по номеру источника прерывания в таблице векторов определяется смещение, запоминается адрес 2 байта в IP и 2 байта в CS. Далее выполняется прерывание по сохранённому адресу и далее восстанавливается информация прерванного процесса и управление возвращается прерванной программе. 

2. Какого типа прерывания использовались в работе?

Int 10h и int 21h – программные прерывания, а также аппаратное прерывание 1Ch.