**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра Математического Обеспечения ЭВМ**

отчет

**по лабораторной работе №4**

**по дисциплине «Операционные системы»**

Тема: **Обработка стандартных прерываний.**

|  |  |  |
| --- | --- | --- |
| Студент гр. 6383 |  | Азаревич А.Д. |
| Преподаватель |  | Губкин А.Ф. |

Санкт-Петербург

2018

# Цель работы

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.

# 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.

Обработчик прерывания - это отдельная процедура, имеющая следующую структуру:

ROUT PROC FAR

PUSH AX ; сохранение изменяемых регистров

…

<действия по обработке прерывания>

POP AX ; восстановление регистров

MOV AL, 20H

OUT 20H, AL

IRET

ROUT ENDP

Две последние строки необходимы для разрешения обработки прерываний с более низкими уровнями, чем только что обработанное. Для установки написанного прерывания в поле векторов прерываний используется функция 25H прерывания 21H, которая устанавливает вектор прерывания на указанный адрес.

PUSH DS

MOV DX, OFFSET ROUT ; смещение для процедуры в DX

MOV AX, SEG ROUT ; сегмент процедуры

MOV DS, AX ; помещаем в DS

MOV AH, 25H ; функция установки вектора

MOV AL, 1CH ; номер вектора

INT 21H ; меняем прерывание

POP DS

Программа, выгружающая обработчик прерываний должна восстанавливать оригинальные векторы прерываний. Функция 35 прерывания 21H позволяет восстановить значение вектора прерывания, помещая значение сегмента в ES, а смещение в BX. Программа должна содержать следующие инструкции:

; -- хранится в обработчике прерываний

KEEP\_CS DW 0 ; для хранения сегмента

KEEP\_IP DW 0 ; и смещения прерывания

; -- в программе при загрузке обработчика прерывания

MOV AH, 35H ; функция получения вектора

MOV AL, 1CH ; номер вектора ШТЕ 21Р

MOV KEEP\_IP, BX ; запоминание смещения

MOV KEEP\_CS, ES ; и сегмента

; -- в программе при выгрузке обработчика прерываний CLI

PUSH DS

MOV DX, KEEP\_IP

MOV AX, KEEP\_CS

MOV DS, AX

MOV AH, 25H

MOV AL, 1CH

INT 21H ; восстанавливаем вектор

POP DS

STI

Для того, чтобы оставить процедуру прерывания резидентной в памяти, следует воспользоваться функцией DOS 31h прерывания 21h. Эта функция оставляет память, размер которой указывается в качестве параметра, занятой, а остальную память освобождает и осуществляет выход в DOS.

Функция 31h int 21h использует следующие параметры:

AH - номер функции 31h;

AL - код завершения программы;

DX - размер памяти в параграфах, требуемый резидентной программе.

Вывод на экран информации обработчиком прерываний осуществляется с помощью функций прерывания 10h:

* Функция 03h – считать позицию курсора в DX (текущие строка и колонка курсора) и в CX (текущие начальная и конечная строки курсора);
* Функция 13h – выводит сообщение, состоящее из CX символов, расположенное в ES:BP на видео страницу BH начиная со строки DH, колонки DL;
* Функция 02h – устанавливает курсор на видеостраницу BH, строку DH, колонку DL.

# Интерфейс функций и структур управляющей программы

Функции:

TETR\_TO\_HEX – переводит число из младшей половины регистра AL в его символьное обозначение (помещается в AL);

BYTE\_TO\_HEX – переводит число из регистра AL в его символьное обозначение (помещается в AX);

WRD\_TO\_HEX – переводит число из регистра AX в его символьное обозначение (помещается в память, на конец которой указывает ES:DI);

INTERRUPT – прерывание, устанавливаемое вместо прерывания системного таймера.

Структуры:

KEY – сигнатура, позволяющая определить, установлено системное или же резидентное прерывание;

INT\_ALR – сообщение о том, что резидентное прерывание уже установлено;

INT\_SET – сообщение о том, что устанавливается резидентное прерывание;

INT\_RES – сообщение о том, что резидентное прерывание заменяется системным;

TCL\_KEY – сигнатура, позволяющая определить соответствует ли хвост командной строки, переданный программе, «ключом» к смене резидентного прерывания системным;

INT\_KEY - сигнатура, позволяющая определить, установлено системное или же резидентное прерывание (принадлежит резидентному прерыванию);

STRING – строка, выводящая на экран количество вызовов резидентного прерывания (принадлежит резидентному прерыванию);

KEEP\_CS – переменная, содержащая сегмент системного прерывания (принадлежит резидентному прерыванию);

KEEP\_IP – переменная, содержащая смещение системного прерывания (принадлежит резидентному прерыванию);

KEEP\_PSP – переменная, содержащая PSP резидента (принадлежит резидентному прерыванию);

COUNTER – счетчик вызова резидентного прерывания (принадлежит резидентному прерыванию);

KEEP\_SP – переменная, сохраняющая SP (принадлежит пользовательскому прерыванию);

KEEP\_SS – переменная, сохраняющая SS (принадлежит пользовательскому прерыванию);

KEEP\_AX – переменная, сохраняющая AX (принадлежит пользовательскому прерыванию);

NEW\_STACK – стек пользовательского прерывания (принадлежит пользовательскому прерыванию).

# Последовательность действий, выполняемых утилитой

Программа определяет расположение системного прерывания в оперативной памяти и определяет, является ли это прерывание системным либо резидентным. Если прерывание системное, то оно заменяется резидентным и программа завершается, оставляя в памяти PSP и прерывание. Иначе, если программа запущена с ключом /un то резидентное прерывание заменяется системным и удаляется из памяти. Иначе программа выводит сообщение о том, что резидентное прерывание установлено, и завершается.

Резидентное прерывание с каждым выводом увеличивает счётчик вызова прерывания и выводится число вызовов этого прерывания.

Пример работы программы указаны на рис. 1-6.

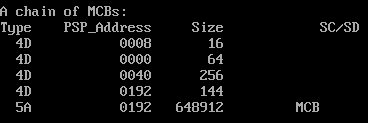


Рис. 1 – Состояние списка MCB до установления резидентного прерывания



Рис. 2 – Установлено резидентное прерывание

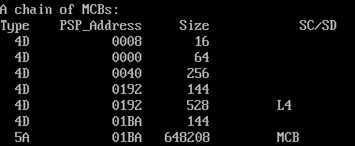


Рис. 3 – Состояние списка MCB после установки резидентного прерывания



Рис. 4 – Попытка запуска программы при установленном резидентном прерывании без ключа /un



Рис. 5 – Попытка запуска программы при установленном резидентном прерывании c ключом /un

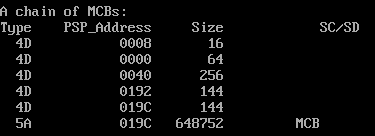


Рис. 6 – Состояние списка MCB после восстановления системного прерывания

# Результаты исследования

1. Сигналы таймера генерируются аппаратурой через определённые интервалы времени. При возникновении такого сигнала возникает прерывание с определённым значением вектора.
2. В работе использовались программные (10h, 21h) и аппаратное (1Ch) прерывания.

# Вывод

В лабораторной работе были изучены принципы установки нового прерывания, оставления его в памяти и последующего возвращения системного прерывания на его «законное» место и удаления «отработавшего» прерывания из памяти.