

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №5
по дисциплине «Организация ЭВМ и систем»
Тема: Разработка собственного прерывания
Вариант 6

Студент гр. 1383

Куликов М.Д.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2022

Цель работы.

Изучение прерываний и разработка собственного прерывания на языке Ассемблера.

Задание.

Вариант 6(1f)

1 - 08h - прерывание от системного таймера - генерируется автоматически операционной системой 18 раз в сек

f - Вывод на экран заданного количества (3-5) сообщений, задержка между которыми возрастает в 2 раза, начиная от 1 сек.

Выполнение работы.

В процедуре Main с помощью функции 35h прерывания 21h мы получаем оригинальный вектор и заменяем прерывание на нужное нам с помощью функции 25h.

После этого мы устанавливаем в регистр CX нужное количество выведенных сообщений, после чего вызываем наше прерывание.

В процедуре обработки прерывания проверяется флаг INT_FLAG на равенство с 1, при неравенстве программа устанавливает этот флаг в 1 и продолжает выполнение и запускает выводы с задержкой. Эти действия обусловлены тем, что прерывание 08h запускается 18 раз в секунду, а вывести сообщения нам нужно определенное количество раз.

После выполнения процедуры обработки прерывания восстанавливается изначальный вектор прерывания и программа завершает свою работу.

Выводы.

В ходе работы были изучены прерывания и было разработано собственное прерывание.

ПРИЛОЖЕНИЕ А
ИСХОДНЫЙ КОД ПРОГРАММЫ

```
AStack SEGMENT STACK
    db 2048 DUP(?)
AStack ENDS

DATA SEGMENT
    KEEP_CS DW 0
    KEEP_IP DW 0
    GREETING DB 'Student from 1383 - Kulikov M.D.', 10, 13, '$'
    COUNTER DB 0
    CUR_DELAY DB 1
    INT_FLAG DB 0
DATA ENDS

CODE SEGMENT
ASSUME CS : CODE, DS : DATA, SS : AStack

ONE_SEC_DELAY Proc NEAR
    push cx
    push dx
    mov cx, 0fh
    mov dx, 4240h
    mov ah, 86h
    int 15h
    pop dx
    pop cx
    ret
ONE_SEC_DELAY ENDP
```

SUBR_INT PROC FAR

MOV BL,1

CMP BL,INT_FLAG

JE THE_END

INC INT_FLAG

PRINT_LOOP:

MOV DX, OFFSET GREETING

mov ah,9

INT 21H

cmp cx, 1

je THE_END

DELAY:

CALL ONE_SEC_DELAY

inc COUNTER

MOV AL, COUNTER

CMP AL,CUR_DELAY

JL DELAY

SHL CUR_DELAY,1

AND COUNTER,0

LOOP PRINT_LOOP

THE_END:

MOV AL, 20H

OUT 20H, AL

IRET

SUBR_INT ENDP

Main PROC FAR

push DS

sub AX, AX

push AX

mov AX, DATA

mov DS, AX

MOV AH, 35H

MOV AL, 08H

INT 21H

MOV KEEP_IP, BX

MOV KEEP_CS, ES

PUSH DS

MOV DX, OFFSET SUBR_INT

MOV AX, SEG SUBR_INT

MOV DS, AX

MOV AH, 25H

MOV AL, 08H

INT 21H

POP DS

MOV CX, 3

INT 08H

CLI

PUSH DS

MOV DX, KEEP_IP

MOV AX, KEEP_CS

MOV DS, AX

MOV AH, 25H

MOV AL, 08H

INT 21H

POP DS

STI

ret

Main ENDP

CODE ENDS

END Main