

БГУИР

Кафедра ЭВМ

Отчет по лабораторной работе №6  
Тема: «Программирование клавиатуры»

Выполнил:  
студент группы 050504 Матусевич С.К.

Проверил:  
Одинец Д.Н.

Минск  
2022

## 1. Постановка задачи

Программируя клавиатуру помогать ее индикаторами. Алгоритм мигания произвольный. Условия реализации программы:

1. Запись байтов команды должна выполняться только после проверки незанятости входного регистра контроллера клавиатуры. Проверка осуществляется считыванием и анализом регистра состояния контроллера клавиатуры.

2. Для каждого байта команды необходимо считывать и анализировать код возврата. В случае считывания кода возврата, требующего повторить передачу байта, необходимо повторно, при необходимости – несколько раз, выполнить передачу байта. При этом повторная передача данных не исключает выполнения всех оставшихся условий.

3. Для определения момента получения кода возврата необходимо использовать аппаратное прерывание от клавиатуры.

4. Все коды возврата должны быть выведены на экран в шестнадцатеричной форме.

## 2. Алгоритм решения

Для вывода на экран скан-кодов или кодов возврата необходимо заменить обработчик прерывания 09h. При вызове данного обработчика выводится значение из порта 60h на экран. При управлении индикаторами значение из порта 60h (код возврата) необходимо анализировать на случай необходимости повторной передачи байтов команды.

В случае считывания кода возврата, требующего повторить передачу байта, устанавливается флаг `isResend`, сигнализирующий о том, что необходимо повторить передачу команды в порт 60h.

Для управления индикаторами клавиатуры используется команда *EDh*. Второй байт этой команды содержит битовую маску для настройки индикаторов (бит 0 – состояние Scroll Lock, бит 1 – состояние Num Lock, бит 2 – состояние Caps Lock). В данной программе управление индикаторами реализовано в функции `void indicator (unsigned char mask)`, где `mask` – битовая маска, определяющая состояние индикаторов.

Перед каждой командой записи происходит ожидание освобождения входного буфера клавиатуры: `while((inp(0x64) & 0x02) != 0x00).`

## 3. Листинг программы

```
#include <io.h>
#include <dos.h>
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>

int exit_f = 0;
```

```

void interrupt(*old9)(...);
void interrupt new9(...) {
    unsigned char c = inp(0x60);
    if (c == 0x01) exit_f = 1;
    if (c != 0xFA && !exit_f) {
        printf("0x%x ", c);
    }
    (*old9)();
}

void indicator(int mask) {
    if (mask != 0xED) indicator(0xED);
    int i = 0;
    while ((inp(0x64) & 2) != 0);
    do {
        i++;
        outp(0x60, mask);
    } while (inp(0x60) == 0xFE && i < 3);
    if (i == 3) {
        cputs("\nError: 0xFE - can't send mask\n");
        exit_f = 1;
    }
}

void main() {
    old9 = getvect(0x09);
    setvect(0x09, new9);
    while (!exit_f) {
        indicator(0x04); // вкл. индикатор Caps Lock
        delay(600);
        indicator(0x00); // выкл. все индикаторы
        delay(600);
        system("cls");
        printf("Press ESC to exit program");
        printf("\nPress any key: ");
    }
    indicator(0x00); // выкл. все индикаторы
    setvect(0x09, old9);
}

```

## 4. Тестирование

Далее приведены результат работы программы:

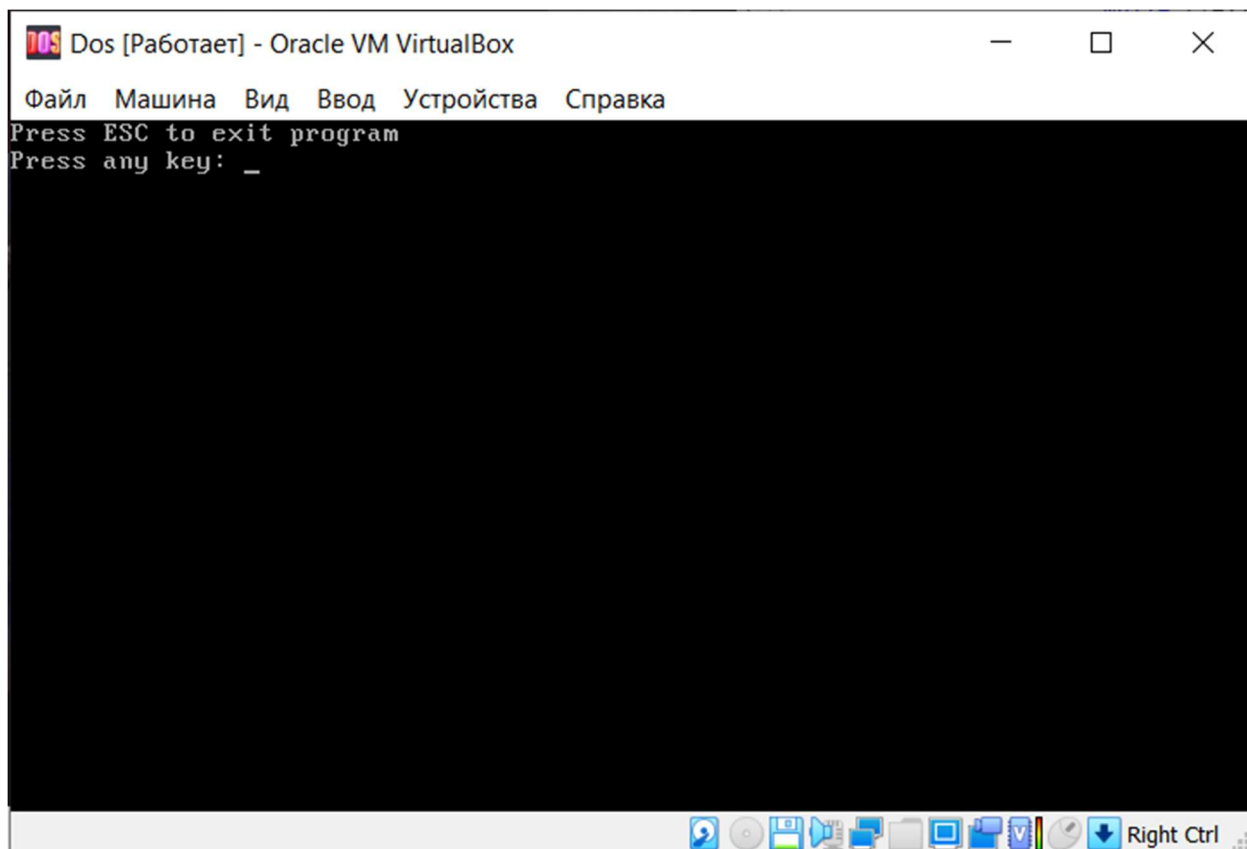


Рисунок 4.1 – Запуск 1

## 5. Заключение

В ходе лабораторной работы была запрограммирована клавиатура, удалось помигать индикаторами, были получены коды возврата и выведены на экран в шестнадцатеричной форме.

Программа компилировалась в Turbo C++ и запускалась в DOS. Для эмуляции DOS используется Oracle Virtual Box на хосте 64-ех разрядной Windows 10.