

Министерство образования Республики Беларусь  
Учреждение образования  
«Белорусский государственный университет информатики и  
радиоэлектроники»

Кафедра электронных вычислительных машин

Лабораторная работа №3  
«Программирование системного таймера»  
Вариант 3

Выполнил:  
Студент группы 250501  
Снитко Д. А.

Проверил:  
Преподаватель  
Одинец Д.Н.

Минск 2024

## 1. Постановка задачи

Задание состоит из двух частей. Первая часть общая для всех. Вторая часть по вариантам.

### **Первая часть** (общее задание):

Запрограммировать второй канал таймера таким образом, чтобы динамик компьютера издавал звуки.

### **Вторая часть:**

1 Для всех каналов таймера считать слово состояния и вывести его на экран в двоичной форме.

2 Для всех каналов таймера рассчитать коэффициент деления (значение счетчика СЕ) и вывести его на экран в шестнадцатеричной форме.

## 2. Алгоритм

Чтобы включить звук на динамике компьютера, нужно выполнить следующие действия:

Записать управляющее слово 10110110 в порт управляющего регистра с адресом 43h. Это управляющее слово соответствует каналу 2 и режиму 3.

Установить значение счетчика канала 2 таймера. Для этого вывести в порт 42h значение, полученное при разделении 1193180 на требуемую частоту в герцах. Сначала вывести младший байт, а затем старший байт значения.

Включить звук, установив в 1 два младших бита порта 61h. Для этого считать байт из порта 61h в рабочую ячейку памяти, установить нужные биты, а затем вывести новое значение байта в порт 61h.

Чтобы выключить звук, установить в 0 два младших бита порта 61h.

Для чтения слова состояния каналов нужно выполнить следующие действия:

Вывести в порт управляющего регистра с адресом 43h управляющее слово, соответствующее команде RBC (Чтение состояния канала) и номеру канала.

Вывести из порта нужного канала слово состояния, чтобы прочесть его.

## 3. Листинг программы

Далее приведен листинг программы, реализующей все поставленные задачи.

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

```

int frequencies[] = { 784, 392, 440, 440, 784, 440, 493, 493 };

int duration[] = 200;

void sound(int frequency, int time) {

// b6 = 10110110 - channel 2(3), операция 11 - чтение/запись младшего, а затем
старшего байта, генерация прямоугольного импульса, 0 - двй код, 43h - регистр команд
    outp(0x43, 0xB6);

    long signal = 1193180; // внутренняя частота таймера
    long cnt = signal / frequency; //Расчет значения счетчика для заданной частоты

    outp(0x42, cnt % 256); // запись младшего байта
    cnt /= 256; // запись старший байта
    outp(0x42, cnt); // запись значения в порт динамика

    //включение динамика, 3 = 00000011
    outp(0x61, inp(0x61) | 3);

    delay(time);

    //выключение динамика, fc = 11111100
    outp(0x61, inp(0x61) & 0xFC);
}

void state_channel() {
    unsigned char temp;
    char* string = new char[9];

    //чтение состояния канала 0
    //11 - режим считывания, 10 - получение состояния канала, 001 - номер канала
    outp(0x43, 0xE2); //E2 = 11100010

    temp = inp(0x40); // получение слова состояния канала
    itoa(temp, string, 2); // integer to ascii переводим значение в строку
    cout << "Channel 0 state word: " << string << endl;

    // канал 1
    outp(0x43, 0xE4); //E4 = 11100100

    temp = inp(0x41);
    itoa(temp, string, 2);
    cout << "Channel 1 state word: " << string << endl;

    // канал 2
    outp(0x43, 0xE8); //E8 = 11101000

    temp = inp(0x42);
    itoa(temp, string, 2);
    cout << "Channel 2 state word: " << string << endl;

    free(string);
}

```

```

void print_counter(int channel) {

    //управляющее слово устанавливает режим чтения счетчика указанного канала
    unsigned char controlWord = 0xE2 | (channel << 1);
    outp(0x43, controlWord);

    unsigned short counter;
    outp(0x43, 0x00); // выбор младшего байта в регистре
    counter = inp(0x40); // Младший байт
    counter |= (inp(0x40) << 8); // Старший байт, сдвиг на 8 разрядов, логическое
    ИЛИ, объединение младшего и старшего байта в одно 16 битное значение

    cout << "Channel " << channel << " counter: " << hex << counter << endl;
}

int main() {
    for (int i = 0; i < 8; i++) {
        sound(frequencies[i], duration[i]);
    }
    state_channel();

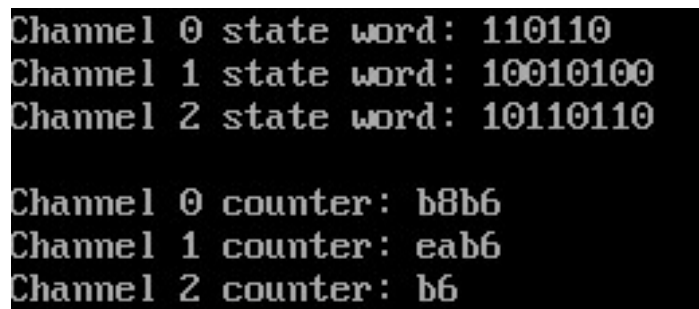
    printf("\n");

    for (int j = 0; j < 3; j++) {
        print_counter(j);
    }
    return 0;
}

```

#### 4. Тестирование программы

Во время работы программы происходит звучание системного динамика. Также для всех каналов таймера выводится на экран в двоичной форме слово состояния и коэффициент деления (значение счетчика CE) в шестнадцатеричной форме:



```

Channel 0 state word: 110110
Channel 1 state word: 10010100
Channel 2 state word: 10110110

Channel 0 counter: b8b6
Channel 1 counter: eab6
Channel 2 counter: b6

```

Рисунок 4.1 – Результат работы программы

## **5. Заключение**

В ходе лабораторной работы удалось запрограммировать второй канал таймера таким образом, чтобы динамик компьютера издавал звуки, для всех каналов таймера было считано слово состояния и выведено на экран в двоичной форме и рассчитан коэффициент деления (значение счетчика СЕ) и выведен на экран в шестнадцатеричной форме.

Программа компилировалась в BorlandC и запускалась в DOS, который эмулировался с помощью DosBox.