

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

Кафедра ЭВМ

Отчет по лабораторной работе №3  
«Программирование системного таймера»

Проверил:  
к.т.н., доцент  
Одинец Д.Н.

Выполнил:  
студент гр. 150501  
Почебут А.С.

Минск 2023

## 1. Постановка задачи

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

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

## 2. Алгоритм

Для того чтобы динамик компьютера издавал звуки, необходимо выполнить следующие действия:

- Вывести в порт управляющего регистра с адресом 43h управляющее слово 10110110, соответствующее каналу 2, режиму 3
- Установить значение счётчика канала 2 таймера: в порт 42h вывести значение, полученное при делении 1193180 на требуемую частоту в герцах, причём вначале вывести младший, а затем старший байты.
- Установить в 1 два младших бита порта 61h для включения звука. Для этого вначале считывается байт из порта 61h в рабочую ячейку памяти, устанавливаются нужные биты, затем выводится новое значение байта в порт 61h.
- Установить в 0 два младших бита порта 61h для выключения звука.

Для чтения слова состояния каналов необходимо:

- 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.

## 3. Листинг программы

Далее приведен листинг программы, выполняющей все поставленные задачи.

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

unsigned int notes[] = { 392,392,392,329,349,349,349,293,293 };
unsigned int note_delay = 400;

void PlaySound();
void StateWords();
void CharToBin(unsigned char state, char* str);
void TurnSpeaker(int isActive);
void SetCount(int iDivider);
void Menu();

int main() {
    Menu();
    return 0;
}
```

```

void Menu() {
    int choice = 0;
    while (1) {
        system("cls");
        printf("1 - Play sound");
        printf("\n2 - Print channels state words");
        printf("\n0 - Exit");

        printf("\n\nEnter choice: ");
        scanf("%d", &choice);
        if (choice >= 0 && choice <= 2) {
            switch (choice) {
                case 0:
                    return;

                case 1:
                    PlaySound();
                    break;

                case 2:
                    StateWords();
                    printf("\n\nPress any key to continue: ");
                    scanf("%d", &choice);
                    break;
            }
        }
    }
}

```

```

void StateWords()
{
    char* bin_state;
    int iChannel;
    unsigned char state;
    bin_state = (char*)calloc(9, sizeof(char));
    if (bin_state == NULL)
    {
        printf("Memory allocation error");
        exit(EXIT_FAILURE);
    }

    for (iChannel = 0; iChannel < 3; iChannel++)
    {
        switch (iChannel)
        {
            case 0:
            {
                outp(0x43, 0xE2);
                state = inp(0x40);
                CharToBin(state, bin_state);
                printf("Channel 0x40 word: %s\n", bin_state);
                break;
            }
            case 1:
            {
                bin_state[0] = '\0';
                outp(0x43, 0xE4);
                state = inp(0x41);

```

```

        CharToBin(state, bin_state);
        printf("Channel 0x41 word: %s\n", bin_state);
        break;
    }
    case 2:
    {
        bin_state[0] = '\0';
        outp(0x43, 0xE8);
        state = inp(0x42);
        CharToBin(state, bin_state);
        printf("Channel 0x42 word: %s\n", bin_state);
        break;
    }
}
}
free(bin_state);
return;
}

```

```

void CharToBin(unsigned char state, char* str)
{
    int i, j;
    char temp;
    for (i = 7; i >= 0; i--)
    {
        temp = state % 2;
        state /= 2;
        str[i] = temp + '0';
    }
    str[8] = '\0';
}

```

```

void SetCount(int iDivider) {
    long base = 1193180;
    long kd;
    outp(0x43, 0xB6);
    kd = base / iDivider;
    outp(0x42, kd % 256);
    kd /= 256;
    outp(0x42, kd);
    return;
}

```

```

void TurnSpeaker(int isActive) {
    if (isActive) {
        outp(0x61, inp(0x61) | 3);
        return;
    }
    else {
        outp(0x61, inp(0x61) & 0xFC);
        return;
    }
}

```

```

void PlaySound() {
    for (int i = 0; i < 9; i++) {
        SetCount(notes[i]);
        TurnSpeaker(1);
    }
}

```

```

        delay(note_delay);
        TurnSpeaker(0);
    }
}

```

#### 4. Тестирование программ

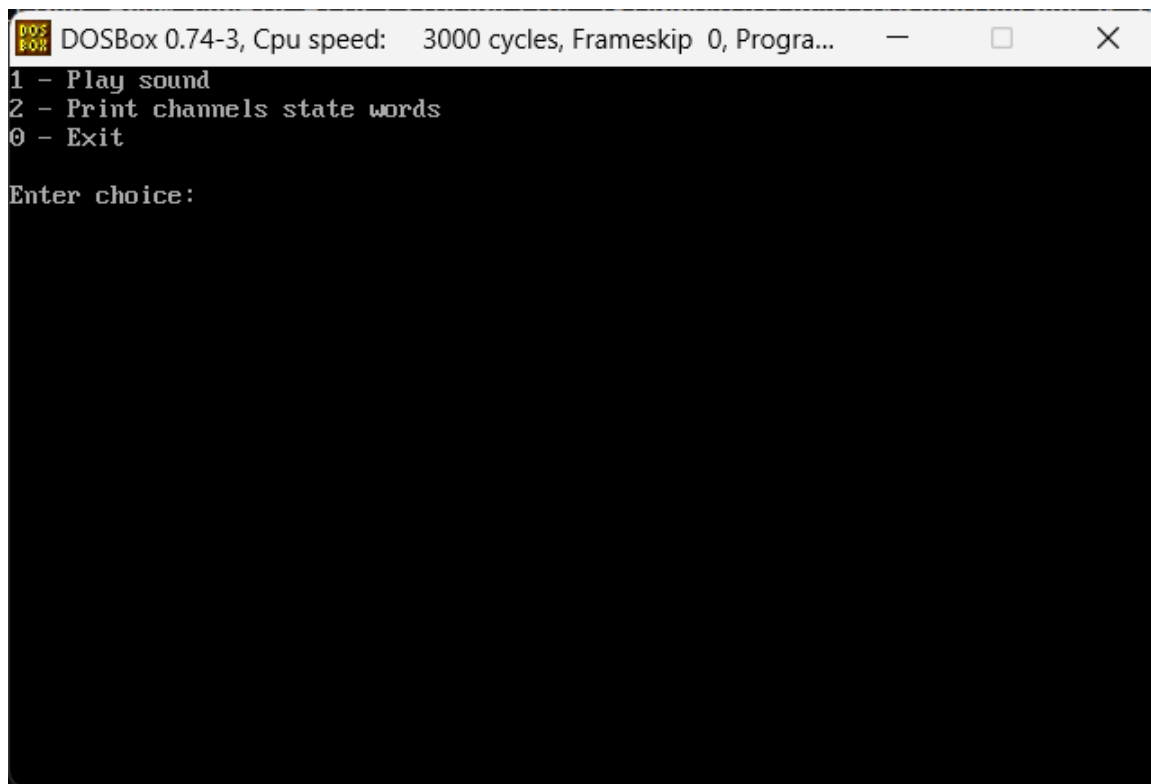


Рисунок 4.1 – Результат работы программы при запуске.

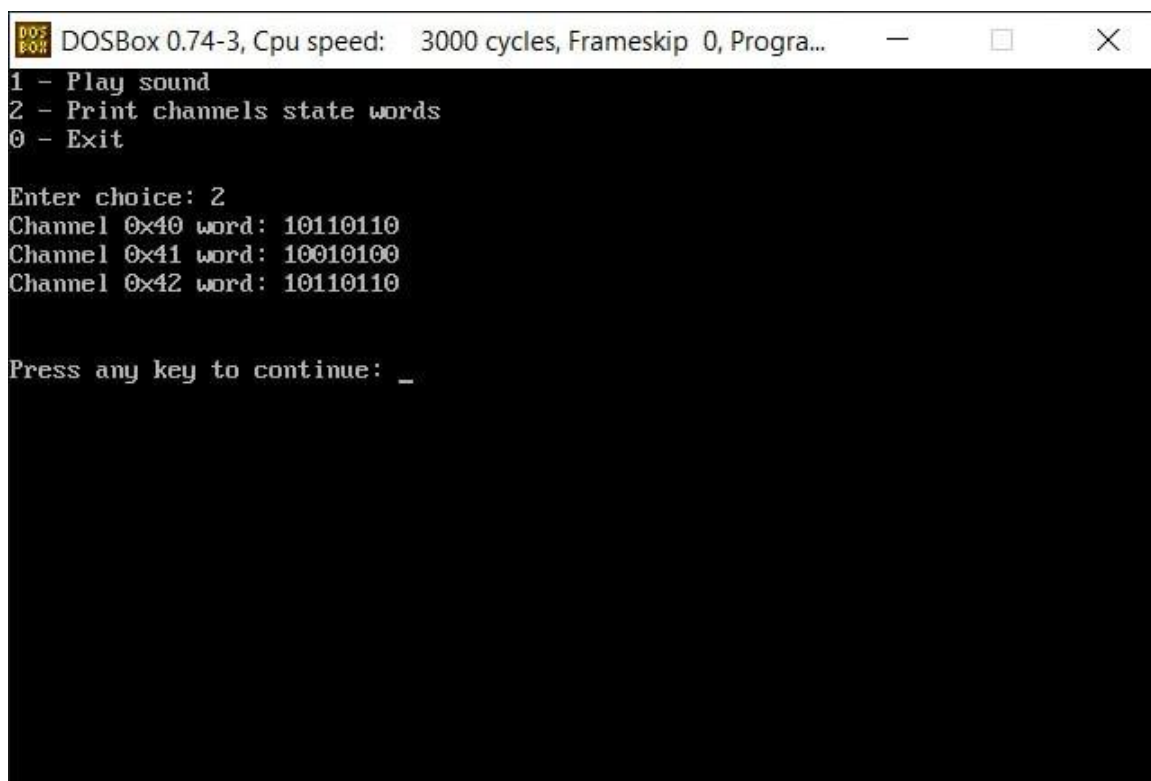


Рисунок 4.2 – Результат работы программы при выводе слов состояний каналов таймера.

## **5. Заключение**

В ходе лабораторной работы удалось запрограммировать второй канал таймера таким образом, чтобы динамик компьютера издавал звук, а также для всех каналов таймера было считано слово состояния и выведено на экран в двоичной форме.

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