УО БГУИР

Кафедра ЭВМ

Отчет по лабораторной работе

Тема: «Системный таймер»

Выполнил:

студент группы 650503 Юревич А.С.

Проверил:

к.т.н., доцент Одинец Д.Н.

Минск 2018

1. Постановка задачи

Под MS DOS написать программу, которая:

1) c помощью системного таймера генерирует звук заданной частоты (по вариантам);

2) выводит слово состояния для каждого канала в двоичном виде;

3) определяет коэффициент деления для каждого канала в 16-ричном виде.

\*Реализовать генератор случайных чисел от нуля до заданного с клавиатуры числа.

Вариант 16:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Частота, Гц (длительность, мс) | | | | | | | | |
| 392  (400) | 392  (400) | 392  (400) | 329  (1000) | 349  (400) | 349  (400) | 349  (400) | 293  (800) | 293  (1000) |

1. Листинг программы

#include <stdio.h>

#include <stdlib.h>

#include <conio.h>

#include <dos.h>

void Menu(void) {

printf("\n1. Play sound");

printf("\n2. Print channels kd");

printf("\n3. Print state words");

printf("\n4. Get random digit");

printf("\n5. Exit.");

printf("\n\nEnter choice: ");

return;

}

void MenuChoice(int \*number) {

for (; ; ) {

while (scanf("%d", number) != 1) {

printf("\nIncorrect input. Enter only digits: ");

while (getchar() != '\n');

continue;

}

if (\*number != 1 && \*number != 2 && \*number != 3

&& \*number != 4 && \*number != 5 && \*number != 6) {

printf("\nIncorrect input. Choose one of points in menu: ");

continue;

}

if (getchar() != '\n') {

printf("\nIncorrect input. Enter only digits: ");

while (getchar() != '\n');

continue;

}

else {

break;

}

}

return;

}

void RandomBoardEnter(long \*number) {

printf("\nEnter random boarder: ");

for (; ; ) {

while (scanf("%ld", number) != 1) {

printf("Incorrect input. Enter only digits: ");

while (getchar() != '\n');

continue;

}

if (\*number <= 0 || \*number >= 65535) {

printf("Incorrect input. Enter right value (less then 65535): ");

continue;

}

if (getchar() != '\n') {

printf("Incorrect input. Enter only digits: ");

while (getchar() != '\n');

continue;

}

else {

break;

}

}

return;

}

void TurnSpeaker(int isActive) {

if (isActive) {

outp(0x61, inp(0x61) | 3);

return;

}

else {

outp(0x61, inp(0x61) & 0xFC);

return;

}

}

void SetSoundFreq(unsigned int freq) {

long base = 1193180; //частоты генератора сигналов таймера

long kd;

outp(0x43, 0xB6);//10110110 канал 2, операция 4, режим 3, формат 0

kd = base / freq;

outp(0x42, kd % 256);

kd /= 256;

outp(0x42, kd);

return;

}

void PlaySound() {

int i;

int c;

printf ("1 - lab melody\n2 - GodFather's Theme\n");

scanf ("%d", &c);

if (c==1)

{

for (i = 0; lab\_notes[i] != 1; i++)

{

SetSoundFreq(lab\_notes[i]);

TurnSpeaker(1);

delay(ndelay[i]);

TurnSpeaker(0);

}

}

if (c==2)

{

for (i = 0; godFather[i] != 1; i++)

{

SetSoundFreq(godFather[i]);

TurnSpeaker(1);

delay(godFatherdelay[i]);

TurnSpeaker(0);

}

}

}

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 PrintKd() {

int iChannel;

long j;

long kd\_low, kd\_high, kd, kd\_max;

for (iChannel = 0; iChannel < 3; iChannel++) {

kd\_max = 0;

for (j = 0; j < 65535; j++) {

switch (iChannel) {

case 0: {

outp(0x43, 0x0);

kd\_low = inp(0x40);

kd\_high = inp(0x40);

kd = kd\_high \* 256 + kd\_low;

break;

}

case 1: {

outp(0x43, 0x40);

kd\_low = inp(0x41);

kd\_high = inp(0x41);

kd = kd\_high \* 256 + kd\_low;

break;

}

case 2: {

outp(0x43, 0x80);

kd\_low = inp(0x42);

kd\_high = inp(0x42);

kd = kd\_high \* 256 + kd\_low;

break;

}

}

if (kd\_max < kd) {

kd\_max = kd;

}

}

switch (iChannel) {

case 0: {

printf("Channel 0x40 kd: %ld\n", kd\_max);

break;

}

case 1: {

printf("Channel 0x41 kd: %ld\n", kd\_max);

break;

}

case 2: {

printf("Channel 0x42 kd: %ld\n", kd\_max);

break;

}

}

}

return;

}

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); //11100010 0x40

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); //11100100 0x41

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); //11101000 0x42

state = inp(0x42);

CharToBin(state, bin\_state);

printf("Channel 0x42 word: %s\n", bin\_state);

break;

}

}

}

free(bin\_state);

return;

}

long RandomDigit() {

long bl, bh;

outp(0x43, 0x86);//4-5 00 6-7 10 2 порт

bl = inp(0x42);

bh = inp(0x42);

bh \*= 256;

bh += bl;

outp(0x61, inp(0x61) & 0xFC);

return bh;

}

void RandTimerSet(long max) {

outp(0x43, 0xB4);//10110100

outp(0x42, max % 256);

max /= 256;

outp(0x42, max);

outp(0x61, inp(0x61) | 1);

return;

}

unsigned int lab\_notes[] = {

392, 392, 392, 329, 349, 349, 349, 293, 293,

392, 392, 392, 329, 349, 349, 349, 293, 293,

392, 392, 392, 329, 349, 349, 349, 293, 293, 1

};

unsigned int ndelay[] = {

400, 400, 400, 1000, 400, 400, 400, 800, 1000,

400, 400, 400, 1000, 400, 400, 400, 800, 1000,

400, 400, 400, 1000, 400, 400, 400, 800, 1000,

};

unsigned int godFather[] = {

196, 262, 311, 294, 262, 311, 262, 294, 262, 208, 247, 196,

196, 262, 311, 294, 262, 311, 262, 294, 262, 196, 185, 175,

175, 208, 247, 294,

175, 208, 247, 262,

131, 156, 233, 208, 196, 233, 208, 208, 196, 196, 123, 131, 1

};

unsigned int godFatherdelay[] = {

400, 400, 400, 400, 400, 400, 400, 400, 400, 400, 400, 1000,

400, 400, 400, 400, 400, 400, 400, 400, 400, 400, 400, 1000,

400, 400, 400, 1500,

400, 400, 400, 1500,

400, 400, 400, 400, 400, 400, 400, 400, 400, 400, 400, 1000,

};

int main() {

int check\_exit = 0;

system("cls");

Menu();

MenuChoice(&check\_exit);

while (check\_exit != 5) {

switch (check\_exit) {

case 1: {

PlaySound();

break;

}

case 2: {

PrintKd();

break;

}

case 3: {

StateWords();

break;

}

case 4: {

long digit;

RandomBoardEnter(&digit);

RandTimerSet(digit);

printf("Press any key to get digit.\n");

getch();

digit = RandomDigit();

printf("\nRandomed digit: %ld\n", digit);

break;

}

}

check\_exit = 0;

system("pause");

system("cls");

Menu();

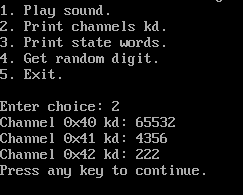
MenuChoice(&check\_exit);

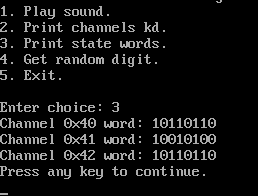
}

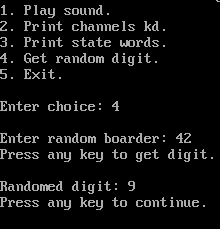
return 0;

}

1. Тестовые пример







1. Заключение

В ходе выполнения работы были получены практические навыки программирования и изучены возможности использования системного таймера 8254, был реализован генератор случайных чисел.