|  |  |
| --- | --- |
|  | **Министерство образования и науки Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана** **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Программное обеспечение ЭВМ и информационные технологии

Отчет по лабораторной работе №3

По курсу: «Архитектура ЭВМ»

По теме: «Проектирование радиоэлектронной аппаратуры на основе

микроконтроллеров ARM7TDMI»

Студент:

Коротков А.В.

Группа:

ИУ7-55Б

Преподаватель:

Попов А.Ю.

Москва, 2019 г.

1. Цель работы

Изучение системы синхронизации микроконтроллера NXP LPC2368 и принципов функционирования таймеров общего назначения.

В ходе работы студенту необходимо ознакомиться с теоретическим материалом, касающимся системы синхронизации и таймеров, разработать и отладить программу функционирования микроконтроллера NXP LPC2368 с использованием отладочных плат SK-LPC2368 и TM1638LED&KEY.

2. Задание

Вариант 8. Устройство управления метрономом, состоящее из двух блоков генерации звука и динамика. Генераторы выдают звуки, соответствующие слабым и сильным долям такта. В каждый момент работает только один генератор. При паузе между сигналами динамик отключен. Программа функционирования: трехдольный ритм. При нажатии на кнопку: отключение.

3. Ход выполнения лабораторной работы

#include <LPC23xx.H>

#define STB 26 //Port1.26

#define CLK 27 //Port1.27

#define DIO 28 //Port1.28

void delay(unsigned int t) {

//Сбросить таймер

T0TC = 0x00000000;

//Установить задержку в мс в регистре совпадения MCR

T0MR0 = t;

//Запустить таймер

T0TCR = 0x00000001;

//Ожидаем окончания счета

while (T0TCR&0x1) {};

}

void tm1638\_sendbyte(unsigned int x) {

unsigned int i;

IODIR1 |= (1<<DIO);//Устанавливаем пин DIO на вывод

for(i = 0; i < 8; i++)

{

IOCLR1=(1<<CLK);//Сигнал CLK устанавливаем в 0

delay(1);//Задержка

if (x&1) {IOSET1=(1<<DIO);} //Устанавливаем значение на выходе DIO

else {IOCLR1=(1<<DIO);}

delay(1);//Задержка

x >>= 1;

IOSET1=(1<<CLK);//Сигнал CLK устанавливаем в 1

delay(2);

}

}

unsigned int tm1638\_receivebyte() {

unsigned int i;

unsigned int x=0;

IODIR1 &= ~(1<<DIO);//Устанавливаем пин DIO на ввод

for(i = 0; i < 32; i++)

{

IOCLR1=(1<<CLK);//Сигнал CLK устанавливаем в 0

delay(1);//Задержка

if (IOPIN1&(1<<DIO)) {

x |= (1<<i);

}

delay(1);//Задержка

IOSET1=(1<<CLK);//Сигнал CLK устанавливаем в 1

delay(2);

}

return x;

}

void tm1638\_sendcmd(unsigned int x)

{

//Устанавливаем пассивный высокий уровень сигнала STB

IOSET1=(1<<STB);

//Устанавливаем пины CLK,DIO,STB на вывод

IODIR1 = (1<<CLK)|(1<<DIO)|(1<<STB);

//Устанавливаем активный низкий уровень сигнала STB

IOCLR1=(1<<STB);

tm1638\_sendbyte(x);

}

void tm1638\_setadr(unsigned int adr) {

//Установить адрес регистра LED инидикации

tm1638\_sendcmd(0xC0|adr);

}

void tm1638\_init() {

unsigned int i;

//Разрешить работу индикации

tm1638\_sendcmd(0x88);

//Установить режим адресации: автоинкремент

tm1638\_sendcmd(0x40);

//Установить адрес регистра LED инидикации

tm1638\_setadr(0);

//Сбросить все

for (i=0;i<=0xf;i++)

tm1638\_sendbyte(0);

//Установить режим адресации: фиксированный

tm1638\_sendcmd(0x44);

}

void Timer0\_Init(void){

//Предделитель таймера = 12000

T0PR = 12000;

//Сбросить счетчик и делитель

T0TCR = 0x00000002;

//При совпадении останавливаем, сбрасываем таймер

T0MCR = 0x00000006;

//Регистр совпадения = 1000 (1 Гц)

T0MR0 = 1000;

}

int main (void) {

unsigned int n = 0,i;

Timer0\_Init(); /\* Настроить таймер \*/

tm1638\_init();/\* Конфигурируем TM1638 \*/

while (1) { /\* Бесконечный цикл \*/

while (i!=0) {

tm1638\_sendcmd(0x46);

i = tm1638\_receivebyte();

}

switch(n)

{

case 0:

tm1638\_setadr(1);

tm1638\_sendbyte(1);

tm1638\_setadr(3);

tm1638\_sendbyte(0);

tm1638\_setadr(5);

tm1638\_sendbyte(1);

break;

case 1:

tm1638\_setadr(1);

tm1638\_sendbyte(0);

tm1638\_setadr(5);

tm1638\_sendbyte(1);

tm1638\_setadr(3);

tm1638\_sendbyte(1);

break;

}

n++;

n %= 2;

delay(100);

tm1638\_setadr(1);

tm1638\_sendbyte(0);

tm1638\_setadr(5);

tm1638\_sendbyte(0);

tm1638\_setadr(3);

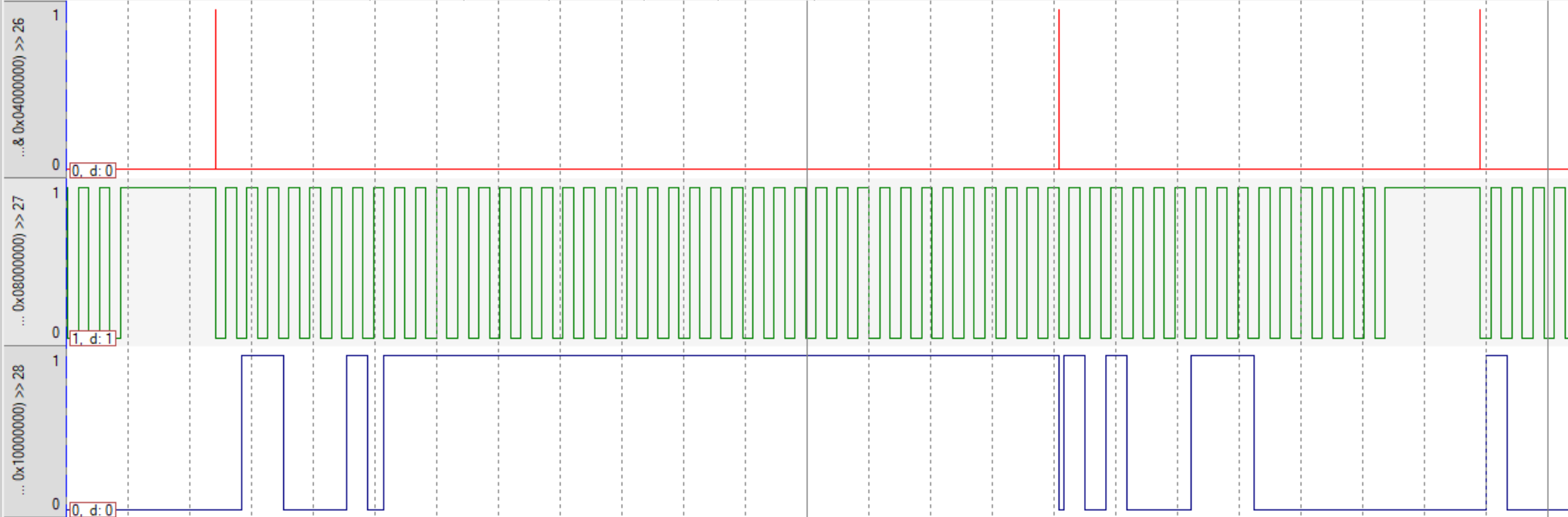
tm1638\_sendbyte(0);

delay(400);

}

}

*Листинг* *1*. *Программа функционирования микроконтроллера.*

**

*Рис 1. Осциллограмма 26-29 битов порта ввода/вывода 1.*

4. Вывод

В ходе лабораторной работы были получены навыки по написанию простейших программ для микроконтроллера NXP LPC2368 на языке C и были освоены базовые навыки для работы в среде «Keil uVISION».