*Федеральное государственное бюджетное образовательное учреждениевысшего профессионального образования*

|  |  |
| --- | --- |
| Gerb-BMSTU_01 | ***«Московский государственный технический университет  имени Н.Э. Баумана (национальный исследовательский институт)»***  ***(МГТУ им. Н.Э. Баумана)*** |

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Программное обеспечение ЭВМ и информационные технологии

**Отчёт**

**по лабораторной работе 2**

**Дисциплина: Архитектура ЭВМ**

**Тема лабораторной работы работы:** Изучение средств ввода и вывода алфавитно-цифровойинформации и индикации с использованием микроконтроллеров ARM7

Студенты гр. ИУ7-51б **\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Сушина А.Д.**

(Подпись, дата) (И.О. Фамилия)

Преподаватель  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Попов А. Ю.**

(Подпись, дата) (И.О. Фамилия)

Москва, 2019г

**Цель работы** – изучение средств управления LED индикаторами и клавиатурными матрицами на базе микросхемы TM1638, а также изучение средств внутрисхемной отладки программ микроконтроллеров ARM7 TDMI. В ходе работы студенту необходимо ознакомиться с особенностями функционирования средств индикации и кнопочных клавиатур на основе микросхемы TM1638, ознакомиться со средствами внутрисхемной отладки программ, разработать и отладить программу индикации и сканирования клавиатуры с использованием отладочной платы SK-LPC2368 и платы индикации TM1638LED&KEY

**Индивидуальное задание**

Устройство состоит из трех исполнительных механизмов и кнопки, подключенных к устройству управления на основе микроконтроллера NXP LPC2368. Разработать программу функционирования микроконтроллера, управляющего работой устройства и обеспечивающую заданную логику его работы:

20. Устройство управления хлебопечкой, состоящее из миксера и двух нагревательных элементов. Программа функционирования:

a) перемешивание и подогрев одним нагревателем;

b) при нажатии на кнопку: выпекание вторым нагревателем.

**Практическая часть**

**Задание 1.** Ознакомиться с теоретическим материалом на стр. 2-14.

**Задание 2.** Доработать программу из лабораторной работы No1 так, чтобы для индикации были использованы светодиоды LED1..LED3 платы TM1638LED&KEY, а также кнопка S1.

**Задание 3.** Разработать и отладить в симуляторе программу функционирования микроконтроллера в соответствии с индивидуальным вариантом.

**Задание 4.** Получить осциллограмму для сигналов STB, CLK и DIO(порты Port1.26, Port1.27, Port1.28)для команды записи регистра управления LED. Осциллограмму и код программы занести в отчет.

**Задание 5.** Выполнить настройку проекта на работу с отладочной платойSK-LPC2368. Выполнить запись информации \*.axf файла проекта в статическую память микроконтроллера.

**Задание 6.** Протестировать правильность функционирования программы с помощью отладочной платы SK-LPC2368. Назначить точку останова. Выполнить пошаговую трассировку программы. Результаты работы программы занести в отчет

**Ход работы**

**Листинг программы функционирования микроконтроллера**

#include <LPC23xx.H> /\* Описание LPC23xx \*/

#define STB 26 //Port1.26

#define CLK 27 //Port1.27

#define DIO 28 //Port1.28

void delay(unsigned int count) {

unsigned int i;

for (i=0;i<count;i++){}

}

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(0xfff);//Задержка

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

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

delay(0xfff);//Задержка

x >>= 1;

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

delay(0x1fff);

}

}

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(0xfff);//Задержка

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

x |= (1<<i);

}

delay(0xfff);//Задержка

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

delay(0x1fff);

}

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);

}

int main (void) {

unsigned int n, i;

tm1638\_init();

while (1) { /\* Бесконечный цикл \*/

for (n = 1; n <= 0xf; n+=2) {

i=1;

while (i!=0) {

tm1638\_sendcmd(0x46);

i = tm1638\_receivebyte();

if (i) {

tm1638\_setadr(1);

tm1638\_sendbyte(0);

tm1638\_setadr(3);

tm1638\_sendbyte(0);

tm1638\_setadr(5);

tm1638\_sendbyte(5);

}

}

tm1638\_setadr(1);

tm1638\_sendbyte(1);

tm1638\_setadr(3);

tm1638\_sendbyte(3);

tm1638\_setadr(5);

tm1638\_sendbyte(0);

}

}

}