Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

«Московский государственный технический университет имени Н.Э. Баумана»

(МГТУ им. Н.Э. Баумана)



## ФАКУЛЬТЕТ \_\_\_\_\_\_\_\_\_\_\_Информатика и системы управления\_\_\_\_\_\_\_\_\_\_

КАФЕДРА Программное обеспечение ЭВМ и информационные технологии

**ОТЧЕТ**

**Лабораторная работа №2**

**По курсу**

**“Архитектура ЭВМ”**

Тема:

“Изучение средств ввода и вывода алфавитно-цифровой информации и индикации с использованием микроконтроллеров ARM7”

Вариант 3

Студент Гасанзаде М.А.

*фамилия, имя, отчество*

Группа ИУ7-56

Москва, 2019

**Оглавление**

[Постановка задачи 3](#_Toc5101)

[Листинг 4](#_Toc5102)

[Вывод 7](#_Toc5103)

[Список литературы 7](#_Toc5103)

# Постановка задачи.

# 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.

# Использованное оборудование: разработка и тестирование проводились в программе “Keil uVision”, которая предоставляет пользователю набор средств для написания и отладки кода программ для микроконтроллеров семейств ARM7, ARM9, Cortex M3 и других.

**Постановка задачи:** 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:

Устройство управления кофеваркой, состоящее из мельницы, нагревателя и клапана кипятка.

Программа функционирования:

a) нагрев воды;

b) при нажатии на кнопку: помол и заварка.

# Листинг

/\* Пример 1.

Управление портами ввода/вывода.

Для правильного связывания модулей отметить

Options -> Linker -> Use Memory Layout from Target Dialog

\*/

#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**);** //Задержка

**}**

delay**(**0xfff**);**

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 flag**,** i**,** mode**;**

tm1638\_init**();**

tm1638\_setadr**(**1**);**

tm1638\_sendbyte**(**1**);**

flag **=** 1**;**

mode **=** 0**;**

**while** **(**flag**)**

**{**

tm1638\_setadr**(**3**);**

tm1638\_sendbyte**(**0**);**

tm1638\_setadr**(**5**);**

tm1638\_sendbyte**(**0**);**

tm1638\_sendcmd**(**0x46**);**

i **=** tm1638\_receivebyte**();**

**if** **(**i **!=** 0**)**

**{**

flag **=** 0**;**

tm1638\_setadr**(**1**);**

tm1638\_sendbyte**(**0**);**

tm1638\_setadr**(**3**);**

tm1638\_sendbyte**(**1**);**

tm1638\_setadr**(**5**);**

tm1638\_sendbyte**(**1**);**

**}**

**}**

**}**

**Выводы**

Устройство функционирует в соответствии с поставленной задачей, успешно «заставили» работать программу не на симуляторе, а на реальном микроконтроллере.

**Список используемой литературы**

1. EVM\_Lab2.pdf

2. Индивидуальны\_ задания 1.pdf