|  |  |
| --- | --- |
|  | **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: «Синхронизация микроконтроллера и управление таймерами»

Студент: Аминов Т.С

Группа: ИУ7-55Б

Преподаватель:

Попов А.Ю.

*Москва, 2019 г.*

**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.

**Задание.**

Вариант 1. Устройство прогрева двигателя внутреннего сгорания, включающее клапан подачи горючей смеси, устройство зажигания, стартер.

Программа функционирования:

1. Пуск стартера, кратковременное открытие клапана горючей смеси на 0.05 секунды и зажигание при закрытом клапане оставшееся время такта (частота: 10 Гц);
2. при нажатии на кнопку: отключение стартера;
3. через 5 секунд после нажатия – отключение зажигания и закрытие клапана;
4. отключение.

Частота внешнего генератора: 12 МГц.

Частота процессорного ядра: 24 МГц.

Частота синхронизации таймера: 12 МГц.

**Листинг программы:**

/\*

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);

}

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;

tm1638\_sendcmd(0x88);//Разрешить работу индикации

//Установить режим адресации: автоинкремент

//Установить адрес регистра LED инидикации

tm1638\_sendcmd(0x40);

tm1638\_setadr(0);//Сбросить адрес

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();

//26 - клапан - 1

//27 - зажигание - 3

//28 - стартер - 5

while (1) /\* Бесконечный цикл \*/

{

i=1;

tm1638\_sendcmd(0x46);

i = tm1638\_receivebyte();

if (i == 1)

{

tm1638\_setadr(1);

tm1638\_sendbyte(0);

tm1638\_setadr(3);

tm1638\_sendbyte(0);

tm1638\_setadr(5);

tm1638\_sendbyte(0);

delay(0xfffff);

}

else

{

tm1638\_setadr(5);

tm1638\_sendbyte(1);

tm1638\_setadr(1);

tm1638\_sendbyte(1);

delay(0xfffff);

tm1638\_setadr(1);

tm1638\_sendbyte(0);

tm1638\_setadr(3);

tm1638\_sendbyte(1);

delay(0xfffff);

tm1638\_setadr(3);

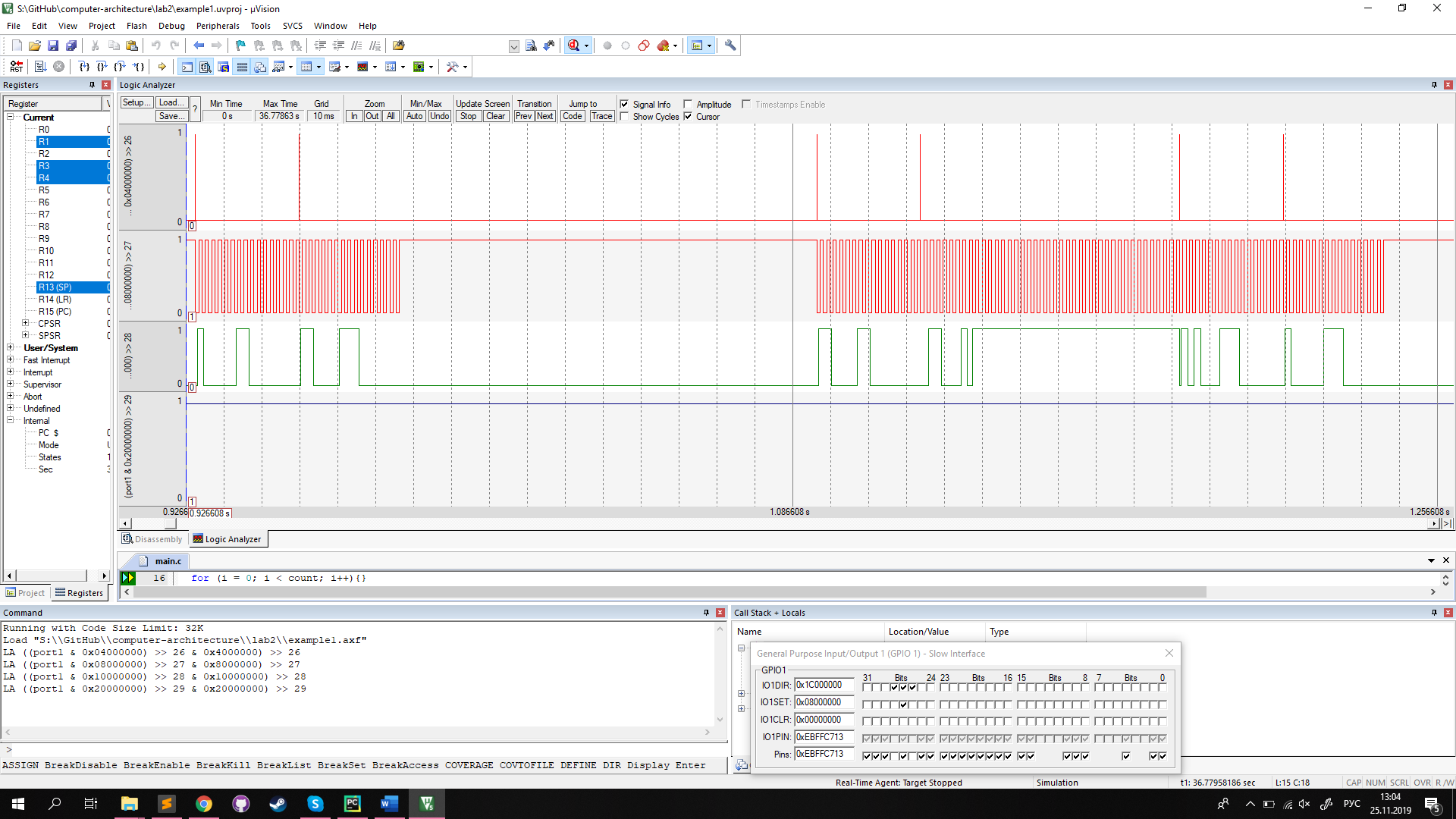
tm1638\_sendbyte(0);

}

}

}

Осциллограмма 26-29 битов порта ввода/вывода 1:



**Вывод.**

В ходе лабораторной работы были получены навыки по написанию простейших программ для микроконтроллеров на базе микросхемы TM1638 на языке C и были освоены базовые навыки для работы в среде Keil uVISION».