



**«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

**О т ч ё т**

**п о л а б о р а т о р н о й р а б о т е 3**

**Дисциплина: Архитектура ЭВМ**

**Тема лабораторной работы:** Синхронизация  
микроконтроллера и управление таймерами

Студенты гр. ИУ7-516 \_\_\_\_\_ Сушина А.Д.  
(Подпись, дата) (И.О. Фамилия)

Преподаватель \_\_\_\_\_ Попов А. Ю.  
(Подпись, дата) (И.О. Фамилия)

Москва, 2019г

**Цель работы** – изучение системы синхронизации микроконтроллера NXPLPC2368 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 NXPLPC2368 с использованием отладочных плат SK-LPC2368 и TM1638LED&KEY

### **Индивидуальные задания.**

Устройство состоит из трех исполнительных механизмов и кнопки, подключенных к устройству управления на основе микроконтроллера NXPLPC2368, подключенного к внешнему генератору синхро сигнала. Разработать программу функционирования микроконтроллера, управляющего работой устройства и обеспечивающую заданную логику его работы при заданных параметрах частоты генератора, частоты процессорного ядра, частоты синхронизации периферии:

20. Устройство управления хлебопечкой, состоящей из миксера и двухнагревательных элементов. Программа функционирования:

- a) перемешивание (2 секунды);
- b) подогрев одним нагревателем до фазы с;
- c) при нажатии на кнопку: выпекание обоими нагревателями (5секунд);
- d) отключение.

Частота внешнего генератора: **12 МГц.**

Частота процессорного ядра: **60 МГц.**

Частота синхронизации таймера: **15 МГц.**

### **Практическая часть**

**Задание 1.** Ознакомиться с теоретическим материалом.

**Задание 2.** Создать проект С программы в среде Keil uVision для микроконтроллера NXP LPC2368 с частотой генератора 12 МГц.

**Задание 3.** Определить параметры M, N, CLKSEL(7:0), PCLKSEL0, PCLKSEL1, обеспечивающие указанные в задании значения частот: Fcpu и Fpclk\_timer0.

**Задание 4.** Разработать и отладить в симуляторе программу функционирования микроконтроллера в соответствии с индивидуальным вариантом. Для

индикации задействовать светодиоды LED1..LED3 платы TM1638LED&KEY, а также кнопка S1 (аналогично лабораторной работе No2).

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

**Задание 6.** Протестировать правильность функционирования программы с помощью отладочных плат SK-LPC2368 и TM1638LED&KEY. Результаты работы в виде таблицы тестов и кода программы занести в отчет.

**Задание 7.** Используя цифровой осциллограф получить осциллограмму одной фазы работы устройства. Определить длительность фазы и определить погрешность полученного временного интервала относительно интервала, указанного в задании. Результаты занести в отчет.

### **Ход работы**

#### **Листинг программы функционирования микроконтроллера**

```
#include <LPC23xx.H>

#define STB 26 //Port1.26
#define CLK 27 //Port1.27
#define DIO 28 //Port1.28

void delay(unsigned int t) {
    T0TC = 0x00000000;
    T0MR0 = t;
    T0TCR = 0x00000001;
    while (T0TCR&0x1) {};
}

void tm1638_sendbyte(unsigned int x) {
    unsigned int i;
    IODIR1 |= (1<<DIO);
    for(i = 0; i < 8; i++)
    {
        IOCLR1=(1<<CLK);
        delay(1);
        if (x&1) {IOSET1=(1<<DIO);}
        else {IOCLR1=(1<<DIO);}
        delay(1);
    }
    x >>= 1;
    IOSET1=(1<<CLK);
}
```

```

        delay(2);
    }
}

unsigned int tm1638_receivebyte() {
    unsigned int i;
    unsigned int x=0;
    IODIR1 &= ~(1<<DIO);
    for(i = 0; i < 32; i++)
    {
        IOCLR1=(1<<CLK);
        delay(1);
        if (IOPIN1&(1<<DIO)) {
            x |= (1<<i);
        }
        delay(1);
        IOSET1=(1<<CLK);
        delay(2);
    }
    return x;
}

void tm1638_sendcmd(unsigned int x)
{
    IOSET1=(1<<STB);
    IODIR1 = (1<<CLK)|(1<<DIO)|(1<<STB);
    IOCLR1=(1<<STB);
    tm1638_sendbyte(x);
}

void tm1638_setadr(unsigned int adr) {
    tm1638_sendcmd(0xC0|adr);
}

void tm1638_init() {
    unsigned int i;
    tm1638_sendcmd(0x88);
    tm1638_sendcmd(0x40);
    tm1638_setadr(0);
    for (i=0;i<=0xf;i++)
        tm1638_sendbyte(0);
    tm1638_sendcmd(0x44);
}

```

```

void Timer0_Init(void){
    T0PR = 12000;
    T0TCR = 0x00000002;
    T0MCR = 0x00000006;
    T0MR0 = 1000;
}

```

```

int main (void) {

    unsigned int n,i;

    Timer0_Init();

    tm1638_init();

    while(1) {
        tm1638_setadr(1);
        tm1638_sendbyte(1);
        delay(2000);
        tm1638_setadr(1);
        tm1638_sendbyte(0);
        while (1) {
            /* Бесконечный цикл */
            i = 1;
            tm1638_sendcmd(0x46);
            i = tm1638_receivebyte();
            if (i) {
                tm1638_setadr(5);
                tm1638_sendbyte(1);
                delay(5000);
                tm1638_setadr(5);
                tm1638_sendbyte(0);
                tm1638_setadr(3);
                tm1638_sendbyte(0);
                break;
            }
            else {
                tm1638_setadr(3);
                tm1638_sendbyte(1);
            }
        }
    }
}

```

## Расчётная часть:

По условию  $F_{CPU} = 60\text{МГц}$ , а  $F_{IN} = 12\text{МГц}$ .

$F_{CPU} = FCCO \cdot CLKSEL(7:0) + 1$ ,  $FCC0 = 2 \cdot M \cdot FINN$ , пусть  $M = 15$ ,  $N = 1$ , тогда  $F_{CC0} = 360\text{ МГц}$ , что удовлетворяет условию  $275\text{ МГц} \leq F_{CC0} \leq 550\text{ МГц}$ , тогда  $CLKSEL(7:0)$  равно 5, что удовлетворяет условию нечетности.

$F_{CLK} = F_{CPU}4$ , поэтому  $PCLKSEL0$  и  $PCLKSEL1$  соответствует пара значений '10'.

<input checked="" type="checkbox"/> Clock Setup	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> System Controls and Status Register (SCS)	
<input checked="" type="checkbox"/> Clock Source Select Register (CLKSRCSEL)	
<input checked="" type="checkbox"/> PLL Configuration Register (PLLCFG)	
MSEL: PLL Multiplier Selection	0x000F
NSEL: PLL Divider Selection	0x0001
<input checked="" type="checkbox"/> CPU Clock Configuration Register (CCLKCFG)	
CCLKSEL: Divide Value for CPU Clock from PLL	0x0006
<input checked="" type="checkbox"/> USB Clock Configuration Register (USBCLKCFG)	
<input checked="" type="checkbox"/> Peripheral Clock Selection Register 0 (PCLKSEL0)	
PCLK_WDT: Peripheral Clock Selection for WDT	Pclk = Cclk / 4
PCLK_TIMER0: Peripheral Clock Selection for TIMER0	Pclk = Cclk / 4
PCLK_TIMER1: Peripheral Clock Selection for TIMER1	Pclk = Cclk / 4
PCLK_UART0: Peripheral Clock Selection for UART0	Pclk = Cclk / 4
PCLK_UART1: Peripheral Clock Selection for UART1	Pclk = Cclk / 4
PCLK_PWM0: Peripheral Clock Selection for PWM0	Pclk = Cclk / 4

Рис 1. Настройки файла LPC2300.s

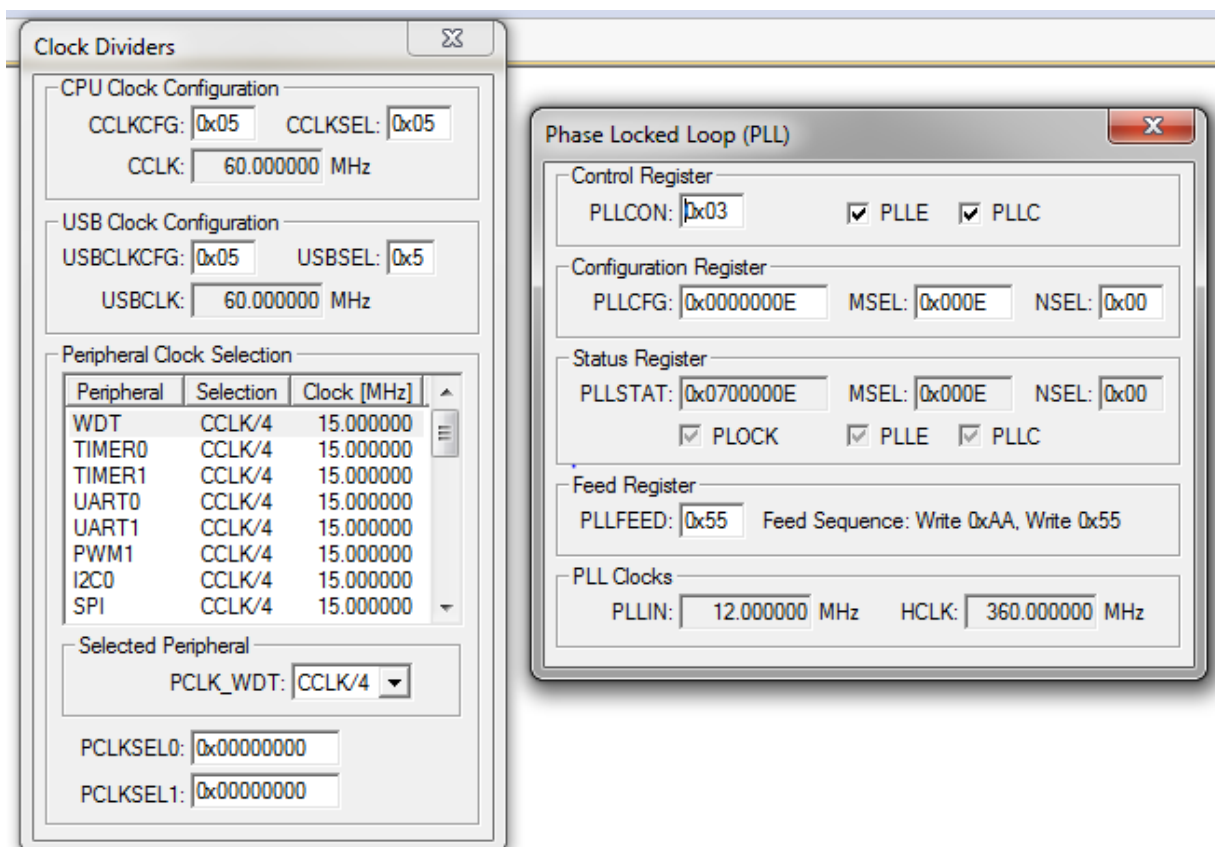


Рис 2. Показатели PLL и Clock dividers

### Результаты тестирования:

При запуске программы загорается первый светодиод(перемешивание) и горит 2 секунды. Затем 1 светодиод выключается и включается светодиод 2(подогрев 1) и ждет нажатия на кнопку. При нажатии на кнопку загорается третий светодиод и горит 5 секунд. Затем оба светодиода отключаются и процесс начинается заново. Это демонстрирует правильность работы хлебопечки.

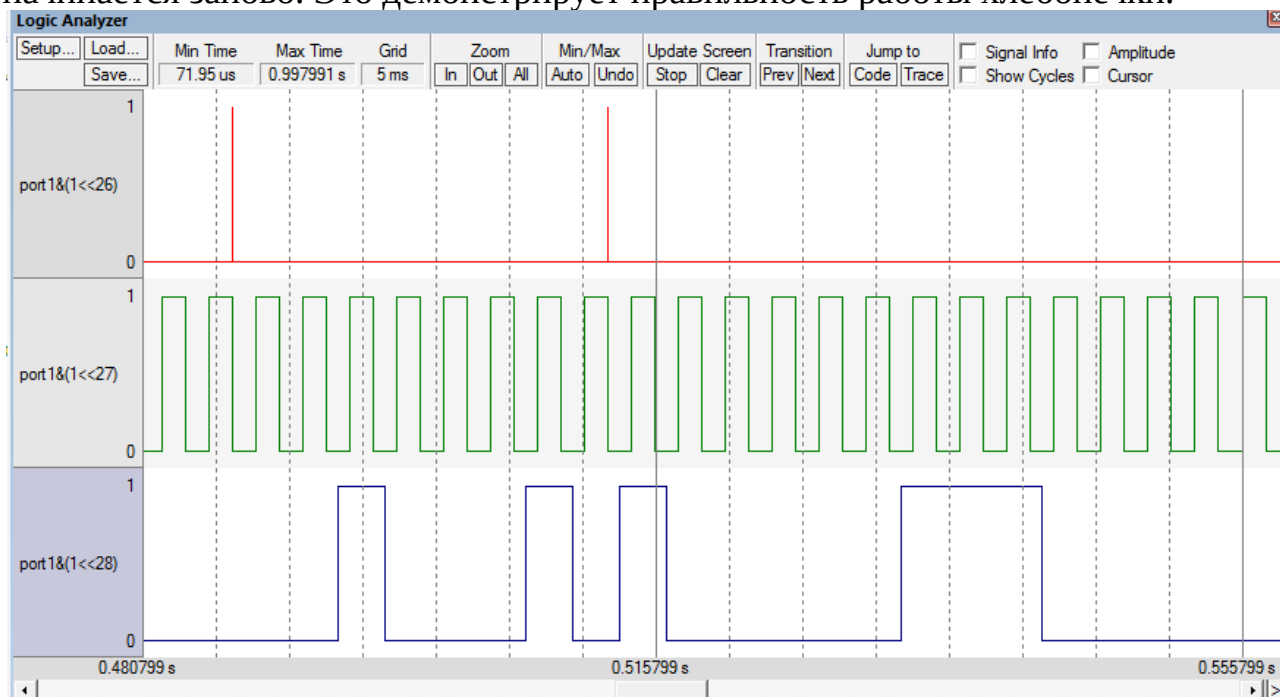


рис 3. Осциллограмма

## **Выводы**

Программа полностью работоспособна и функционирует четко в соответствии с поставленной задачей.





