

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Федеральное государственное автономное образовательное учреждение
высшего профессионального образования**

Санкт-Петербургский государственный университет
аэрокосмического приборостроения

Кафедра № 11 Аэрокосмических измерительно-вычислительных комплексов

Составитель доц. к.т.н. Малаханов Р.Н.

Модуль системного таймера SysTick

Контрольная работа для студентов заочной формы обучения
по дисциплине «Цифровые вычислительные устройства и
микропроцессоры»

Санкт-Петербург

2019

Список сокращений

ДОП – демонстрационно-отладочная плата

ЕИ – единичный индикатор

МК – микроконтроллер

ПО – программное обеспечение

РЦВВ – режим циклического включения-выключения

СИД – светоизлучающий диод

Цель работы заключается в ознакомлении с функциональным устройством модуля системного таймера SysTick микроконтроллера 1986BE93У производства АО «ПКК Миландр» и принципами его программного управления.

Описание модуля системного таймера SysTick приведено в спецификации на микроконтроллеры серии 1986BE9х [1] на страницах с 136 по 139.

Для работы с модулем SysTick предназначены следующие функции.

1. Функция настройки и запуска

```
static uint32_t SysTick_Config(uint32_t ticks).
```

Функция производит запись входного параметра ticks в регистр LOAD, после чего запускает системный таймер SysTick.

Функция возвращает значение 0. Если же значение ticks больше значения $2^{24}-1$, то функция возвращает значение 1 и запуск системного таймера не производится.

2. Функция обработки прерывания SysTick

```
void SysTick_Handler(void).
```

После каждого срабатывания таймера вызывается эта функция.

Расчёт значения для регистра LOAD

Тактовая частота микроконтроллера составляет $F = 8$ МГц; период $T = 1/F$. Длительность временного интервала срабатывания таймера SysTick $\tau = T \cdot \text{LOAD}$, откуда

$$\text{LOAD} = \tau/T = \tau \cdot F.$$

Порядок выполнения контрольной работы

1. В соответствии со своим вариантом разработайте алгоритм работы программы для МК. Варианты индивидуальных заданий приведены в табл. 1. Номер варианта совпадает с последней цифрой в зачётной книжке; если последней цифрой является нуль, то следует выполнять 10 вариант.

2. В соответствии с разработанным алгоритмом разработайте программу на языке программирования Си.

3. Составьте отчёт о выполнении контрольной работы.

ПО следует разрабатывать ДОП 1986EvBrd_48, описание которой приведено в [2]. Электрическая принципиальная схема ДОП приведена в [3].

На ДОП два СИД, выполняющих функции ЕИ и обозначенных как VD2 и VD3 [3], подключены к выводам PF0 и PF1 порта F МК соответственно (сигнальные цепи PF0_VD и PF1_VD). Для программного управления СИД выводы PF0 и PF1 должны быть настроены на работу в режиме цифрового вывода. После этого для включения ЕИ надо в первый или нулевой разряды регистра RXTX записать единицу, а для выключения – нуль.

На ДОП установлены пять кнопок управления. Названия кнопок, а также выводов МК, к которым они подключены, приведены в табл. 2 [5]. При нажатии кно-

пок активным является низкий логический уровень. На электрической принципиальной схеме кнопка *SELECT* обозначена как *SEL*.

Таблица 1 - Варианты заданий

Вар.	Управление состоянием единичных индикаторов VD2 и VD3с помощью кнопок									
	Left		Right		Up		Down		Select	
	VD1	VD2	VD1	VD2	VD1	VD2	VD1	VD2	VD1	VD2
1	Н, 1 Гц	Вкл	К			Н, 2 Гц	Вкл	К		Выкл
2	Н, 1 Гц	Н, 0,5 Гц	К	К	Вкл		Вкл			Выкл
3		Вкл/Выкл		Вкл	Н, 2 Гц	Н, 0,5 Гц	К	К		Выкл
4	Вкл/Выкл			Вкл/Выкл		К	Н, 1 Гц		К	Н, 0,5 Гц
5	Вкл/Выкл			Вкл/Выкл	Н, 2 Гц	Н, 1 Гц	К			К
6		Вкл/Выкл	Вкл/ Выкл		Н, 0,5 Гц	Н, 1 Гц		К	К	
7	Вкл/Выкл			Вкл/Выкл	Н, 1 Гц	Н, 0,5 Гц	К	К		
8	Н, 1 Гц		К		Н, 2 Гц			Вкл	К	Выкл
9		Н, 1 Гц	Вкл/ Выкл	К	Н, 2 Гц		К			Вкл/ Выкл
10	Н, 1 Гц		К		Н, 2 Гц	Н, 0,5 Гц	К	К		
11	Вкл/Выкл			Вкл/Выкл	Н, 0,5 Гц			Н, 0,5 Гц		К

Таблица 2 – Название кнопок и выводов МК

Название кнопки	Вывод МК
<i>SELECT</i>	<i>PC0</i>
<i>UP</i>	<i>PD0</i>
<i>DOWN</i>	<i>PD1</i>
<i>LEFT</i>	<i>PE3</i>
<i>RIGHT</i>	<i>PE6</i>

Буква Н в табл. 1 обозначает начало РЦВВ единичного индикатора, а буква К - соответственно конец этого режима. В этом режиме ЕИ включается и выключается через заданные в таблице интервалы времени (мигает). Обозначение «Вкл/Выкл» указывает, что после каждого нажатия кнопки указанный ЕИ должен изменить своё состояние на противоположное: если он включен, то после нажатия кнопки он должен выключиться и наоборот.

В качестве примера в листинге 1 приведена программа, которая выполняет вариант № 11 в табл. 1.

Листинг 1. Программа с использованием системного таймера SysTick

Программа реализует 11 вариант.

```
#include "MDR32F9Qx_port.h"
#include "MDR32F9Qx_rst_clk.h"

#define          COUNT    4000000

// LED0 - PF0
// LED1 - PF1
// Select - PC0
// Up - PD0
// Down - PD1
// Left -    PE3
// Right-    PE6

enum Key {None, Up, Down, Left, Right, Select};

unsigned gTick;

enum Key GetKey(void);

int main(void)
{

    static PORT_InitTypeDef PortInit;

    // Разрешение тактирования портов
    RST_CLK_PCLKcmd(RST_CLK_PCLK_PORTC | RST_CLK_PCLK_PORTD |
RST_CLK_PCLK_PORTE | RST_CLK_PCLK_PORTF, ENABLE);

    // Настройка выводов RD0 и RD1
    PortInit.PORT_Pin  = (PORT_Pin_0 | PORT_Pin_1);
    PortInit.PORT_OE   = PORT_OE_IN;
    PortInit.PORT_FUNC = PORT_FUNC_PORT;
    PortInit.PORT_MODE = PORT_MODE_DIGITAL;
    PortInit.PORT_SPEED = PORT_SPEED_SLOW;
    PORT_Init(MDR_PORTD, &PortInit);
```

```

// Настройка вывода RC0
PortInit.PORT_Pin = (PORT_Pin_0);
PORT_Init(MDR_PORTC, &PortInit);

// Настройка выводов RE3 и RE6
PortInit.PORT_Pin = (PORT_Pin_3 | PORT_Pin_6);
PORT_Init(MDR_PORTE, &PortInit);

// Настройка выводов RF0 и RF1
PortInit.PORT_Pin = (PORT_Pin_0 | PORT_Pin_1);
PortInit.PORT_OE = PORT_OE_OUT;
PORT_Init(MDR_PORTF, &PortInit);

// Запуск SysTick
SysTick_Config(COUNT);

while(1)
{
    static unsigned char LED_Up=0, LED_Down=0, BlinkUp=0, Blink-
Down=0;
    enum Key key;

    key = GetKey();
    if(key == Left)
    {
        BlinkUp = 0; LED_Up++;
    }

    // Right Key
    if(key == Right)
    {
        BlinkDown = 0; LED_Down++;
    }

    // Up Key
    if(key == Up)
    {
        BlinkUp = 1; LED_Up++; gTick = 0;
        SysTick_Config(COUNT);
    }

    // Down Key
    if(key == Down)

```

```

    {
        BlinkDown = 1; LED_Down++; gTick = 0;
        SysTick_Config(COUNT);
    }

// Select Key
if(key == Select)
{
    BlinkDown = BlinkUp = gTick = 0;
    // Остановка SysTick
    SysTick->CTRL &=~ SysTick_CTRL_ENABLE_Msk;
}

if(gTick)
{
    if(BlinkUp)
        LED_Up++;
    if(BlinkDown)
        LED_Down++;
    gTick = 0;
}

// LED
if(LED_Up & 0x01)
    PORT_SetBits(MDR_PORTF, PORT_Pin_0);
else
    PORT_ResetBits(MDR_PORTF, PORT_Pin_0);

if(LED_Down & 1)
    PORT_SetBits(MDR_PORTF, PORT_Pin_1);
else
    PORT_ResetBits(MDR_PORTF, PORT_Pin_1);
}
}

// =====
enum Key GetKey(void)
{
    if (!PORT_ReadInputDataBit(MDR_PORTE,PORT_Pin_3))
    {
        while (!PORT_ReadInputDataBit(MDR_PORTE,PORT_Pin_3));
        return Left;
    }
    if (!PORT_ReadInputDataBit(MDR_PORTE,PORT_Pin_6))

```

```

        {
            while ( ! PORT_ReadInputDataBit(MDR_PORTE,PORT_Pin_6) );
            return Right;
        }
    if (!PORT_ReadInputDataBit(MDR_PORTD,PORT_Pin_0))
        {
            while (!PORT_ReadInputDataBit(MDR_PORTD,PORT_Pin_0) );
            return Up;
        }
    if (!PORT_ReadInputDataBit(MDR_PORTD,PORT_Pin_1))
        {
            while (!PORT_ReadInputDataBit(MDR_PORTD,PORT_Pin_1) );
            return Down;
        }
    if (!PORT_ReadInputDataBit(MDR_PORTC, PORT_Pin_0))
        {
            while (!PORT_ReadInputDataBit(MDR_PORTC,PORT_Pin_0) );
            return Select;
        }
    return None;
}
// =====
void SysTick_Handler(void)
{
    gTick = 1;
}

```

Нулевые разряды переменных LED_Up и LED_Down предназначены для хранения состояния ЕИ VD2 и VD3: если нулевой разряд установлен, то ЕИ включен; иначе - выключен. Инкремент любой переменной приводит к инверсии её нулевого разряда.

Переменные *BlinkUp* и *BlinkDown* предназначены для указания работы единичных индикаторов VD2 и VD3 в РЦВВ. Если значение какой-нибудь переменной равно единице, то соответствующий ЕИ работает в РЦВВ.

Функция enum Key GetKey(void) возвращает константу из перечисления enum Key, которая соответствует нажатой пользователем кнопке. Если ни одна из кнопок не нажата, то функция возвращает значение None.

Считывание состояния кнопки осуществляется функцией *PORT_ReadInputDataBit()*. При нажатии кнопки пользователь держит кнопку нажа-

той около 0,5 с. За это время основной рабочий цикл МК будет выполнен множество раз и при использовании только вышеназванной функции будет происходить многократное выполнение операторов в конструкции оператора условного перехода *if(!PORT_ReadInputDataBit())* на одно нажатие кнопки. Во избежание этого добавлен пустой цикл *while(!PORT_ReadInputDataBit());*, который предназначен для ожидания момента отпускания кнопки пользователем после её нажатия, в результате чего достигается однократное выполнения операторов, входящих в конструкцию оператора условного перехода *if(!PORT_ReadInputDataBit())*, после одного нажатие кнопки.

Переменная *gTick* предназначена для указания того, что прерывание из-за срабатывания таймера *SysTick* уже произошло.

Оформление отчёта

Отчёт должен быть оформлен в соответствии с требованиями нормоконтроля и должен содержать:

- цель работы;
- задание;
- блок-схему алгоритма программы;
- текст программы на языке программирования Си;
- библиографический список;
- выводы о проделанной работе.

Библиографический список

1. Микросхема 32-разрядного однокристалльного микро-ЭВМ с памятью Flash-типа 1986VE9ху, К1986VE9ху, К1986VE92QI, К1986VE92QC, 1986VE91H4 , К1986VE91H4, 1986VE94H4, К1986VE94H4 [Электронный ресурс]: Специфика-

ция. – Версия 3.10.0 от 04.10.2016. – Электрон. дан. (7,54 Mbytes). – [Б.м.]: АО «ПКК Миландр», 2016. – Режим доступа: <http://ic.milandr.ru/upload/iblock/33e/33e6826dd444292426050ff6e39095a1.pdf>, свободный. – Загл. с экрана.

2. Демонстрационно-отладочная плата 1986*EvBrd_48* [Электронный ресурс] : Техническое описание. – Версия 1.0 от 25.05.2010. – Электрон. дан. (1,05 Mbytes). – [Б.м.]: ЗАО «ПКК Миландр», 2010. – 1 электрон. опт. диск (CD-ROM). – Систем. требования: Pentium 100МГц ; 16 Мб RAM ; Windows 7 ; CD-ROM дисковод ; SVGA видеокарта, 256 цв. ; мышь. - Загл. с экрана. - CD-ROM входит в комплект поставки демонстрационно-отладочной платы 1986*EvBrd_48*.

3. Отладочная плата 1986BE93У [Электронный ресурс] : Схема электрическая принципиальная. – Revision 3. Последнее изменение 24.04.2014. – Электрон. граф. дан. (148 Kbytes). – [Б.м.]: ЗАО «ПКК Миландр», 2014. - 1 электрон. опт. диск (CD-ROM). – Систем. требования: Pentium 100МГц ; 16 Мб RAM ; Windows 7 ; CD-ROM дисковод ; SVGA видеокарта, 256 цв. ; мышь. – Загл. с экрана. - CD-ROM входит в комплект поставки демонстрационно-отладочной платы 1986*EvBrd_48*.