

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.01 Информатика и вычислительная техника**

по лабораторной работе № 4

**Дисциплина:** Микропроцессорные системы.

Преподаватель	Бычков Б. И.
(Подпись, дата)	(И.О. Фамилия)

Москва, 2022

## Вариант 2.

### Цели работы:

- изучение структур и режимов работы таймеров и их программирование;
- анализ схем включения таймеров для проведения исследований;
- программирование задач с таймером.

### Ход работы.

#### Задание 1. Режим счетчика

Проверить на плате STK500 работу заданной программы. Изменить программу, исключив влияние на работу таймера возможность “дребезга” кнопки.

Программа задействует таймер/счетчик T0, структурная схема которого приведена на рисунке 1.

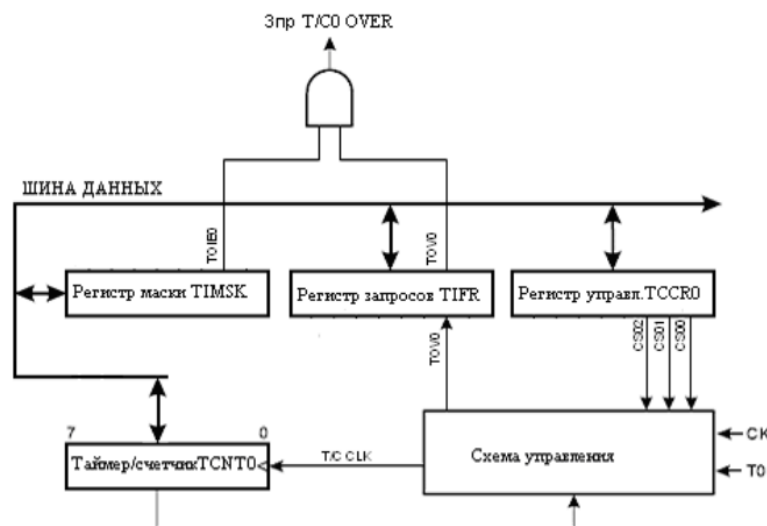


Рис.1. Структурная схема таймера/счётчика T0

### Рисунок 1 – структурная схема таймера/счетчика T0

Код измененной программы:

```
*****
;
;Программа 4.1 для МК ATx8515:
;демонстрация работы таймера T0 в режиме счётчика событий;
;событие - нажатие кнопки SW0.
;Соединения: порт PB0–SW0, шлейфом порт PD-LED
;Светодиоды включаются после четвертого нажатия кнопки SW0
*****
;include "8515def.inc" ;файл определений AT90S8515
```

```

.include "m8515def.inc" ;файл определений ATmega8515
.def temp = r16 ;временный регистр
;***Таблица векторов прерываний
.org $000
rjmp INIT ;обработка сброса
.org $007
rjmp T0_OVF ;обработка переполнения таймера T0
;***Инициализация МК

INIT: ldi temp,low(RAMEND) ;установка
out SPL,temp ; указателя стека
ldi temp,high(RAMEND) ; на последнюю
out SPH,temp ; ячейку ОЗУ
clr temp ;инициализация выводов порта PB
out DDRB,temp ; на ввод
ldi temp,(1<<PB0) ;включение 'подтягивающего' резистора
out PORTB,temp ; входа PB0
ser temp ;инициализация выводов порта PD
out DDRD,temp ; на вывод
out PORTD,temp ;выключение светодиодов
ldi temp,(1<<SE) ;разрешение перехода
out MCUCR,temp ; в режим Idle
;***Настройка таймера T0 на режим счётчика событий
ldi temp,0x02 ;разрешение прерывания по
out TIMSK,temp ; переполнению таймера T0
ldi temp,0x07 ;переключение таймера
out TCCR0,temp ; по положительному перепаду напряжения
sei ;глобальное разрешение прерываний
ldi temp,0xFC ;$FC=-4 для
out TCNT0,temp ; отсчёта 4-х нажатий

LOOP:

sbic PINB,1

rjmp LOOP

cbi PORTB,0

sbi PORTB,0

rcall DELAY

```

```

no: sbis PINB,1

rjmp no

rjmp LOOP

;***Обработка прерывания при переполнении таймера T0
T0_OVF: clr temp
out PORTD,temp ;включение светодиодов
rcall DELAY ;задержка
ser temp
out PORTD,temp ;выключение светодиодов
ldi temp,0xFC ;перезагрузка
out TCNT0,temp ; TCNT0
reti

;*** Задержка ***
DELAY: ldi r19,6
ldi r20,255
ldi r21,255

dd: dec r21
brne dd
dec r20
brne dd
dec r19
brne dd
ret

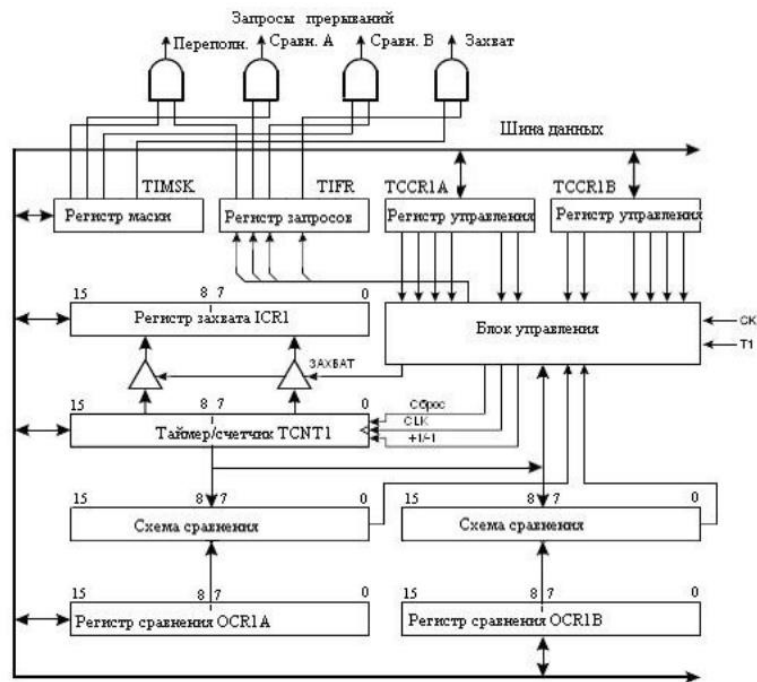
```

Данная программа после считывания нажатия кнопки ожидает некоторое время, чтобы прекратился дребезг. Таймер T0 используется в качестве счетчика событий нажатия на кнопку.

## **Задание 2. Режим таймера**

Проверить работу программы. Оценить время свечения светодиодов при нажатии кнопки SW0 и при нажатии кнопки SW1 и сравнить его с расчетным значением. Изменив настройки таймера, уменьшить вдвое время включения светодиодов.

Схема используемого таймера/счетчика T1 приведена на рисунке 2.



**Рисунок 2 – структурная схема таймер/счетчика T1**

$$T = (65536 - TCNT1) * K / F_{ck}$$

В исходной программе  $TCNT1 = 0x8000 = 32768$ .

$$T_1 = (65536 - 32768) * K / F_{ck} = 32768 * K / F_{ck}$$

В измененной программе:

$$\text{По заданию } T_{1\_нов} = T_1 / 2 = 16384 * K / F_{ck} = (65536 - TCNT1) * K / F_{ck}$$

$$TCNT1 = 65536 - 16384 = 49152 = 0xC000$$

Код измененной программы:

```

;*****
;
;include "8515def.inc" ;файл определений AT90S8515
.include "m8515def.inc" ;файл определений ATmega8515
.def temp = r16 ;временный регистр
.equ SW0 = 0 ;0-ой вывод порта PD
.equ SW1 = 1 ;1-ий вывод порта PD
;***Таблица векторов прерываний
.org $000

rjmp INIT ;обработка сброса
.org $006
rjmp T1_OVF ;обработка переполнения таймера T0

;***Инициализация МК

```

```

INIT: ldi temp,low(RAMEND) ;установка
out SPL,temp ; указателя стека
ldi temp,high(RAMEND) ; на последнюю
out SPH,temp ; ячейку ОЗУ
clr temp ;инициализация выводов порта PD
out DDRD,temp ; на ввод
ldi temp,0x03 ;включение 'подтягивающих' резисторов
out PORTD,temp ; в разрядах 0,1 порта PD
ser temp ;инициализация выводов порта PB
out DDRB,temp ; на вывод
out PORTB,temp ;выключение светодиодов

;***Настройка таймера T1 на режим таймера
ldi temp,0x80 ;разрешение прерывания по
out TIMSK,temp ; переполнению таймера T1
clr temp ;таймер T1
out TCCR1B,temp ; остановлен
ldi temp,0xC0 ; загрузка TCNT1
out TCNT1H,temp
ldi temp,0x00
out TCNT1L,temp
sei ;глобальное разрешение прерываний

;***Ожидание нажатия кнопок
test_sw0: sbic PIND,SW0 ;проверка нажатия
rjmp test_sw1 ; кнопки SW0

;***Обработка нажатия кнопки SW0
ldi temp,0x05 ;для настройки предделителя (K=1024)
rcall LED_ON ;включение светодиодов
test_sw1: sbic PIND,SW1 ;проверка нажатия
rjmp test_sw0 ; кнопки SW1

;***Обработка нажатия кнопки SW1
ldi temp,0x04 ; для настройки предделителя (K=256)
rcall LED_ON ;включение светодиодов
rjmp test_sw0

;***Включение светодиодов
LED_ON: out TCCR1B,temp ;запуск таймера с предделителем
clr temp ;включение
out PORTB,temp ; светодиодов
ret

```

```

;***Обработка прерывания при переполнении таймера T1
T1_OVF: ser temp
out PORTB,temp ;выключение светодиодов
clr temp ;останов
out TCCR1B,temp ; таймера T1
ldi temp,0xC0;
out TCNT1H,temp ; перезагрузка TCNT1
ldi temp,0x00
out TCNT1L,temp
reti

```

Расчетное время свечения светодиодов в исходной программе:

$$T_{sw0} = ((65536 - 32768) * 256) / (3.69 * 10^6) = (32768 * 256) / (3.69 * 10^6) = 2.27 \text{ c}$$

$$T_{sw1} = ((65536 - 32768) * 1024) / (3.69 * 10^6) = (32768 * 1024) / (3.69 * 10^6) = 9.09 \text{ c}$$

Практическое время свечения светодиодов в исходной программе:

$$T_{sw0} = 2.33 \text{ c}$$

$$T_{sw1} = 8.82 \text{ c}$$

Расчетное время свечения светодиодов в измененной программе:

$$T_{sw0} = ((65536 - 32768) * 256) / (3.69 * 10^6) = (32768 * 256) / (3.69 * 10^6) = 1.14 \text{ c}$$

$$T_{sw1} = ((65536 - 32768) * 1024) / (3.69 * 10^6) = (32768 * 1024) / (3.69 * 10^6) = 4.55 \text{ c}$$

Практическое время свечения светодиодов в измененной программе:

$$T_{sw0} = 1.22 \text{ c}$$

$$T_{sw1} = 4.46 \text{ c}$$

Как видно, практически измеренные промежутки времени близки к теоретическим, особенно учитывая высокие относительные погрешности ввиду малого интервала измерения.

### Задание 3. Функция сравнения

Проверить работу программы. Изменить параметры настройки таймера так, чтобы параметры выходных сигналов соответствовали выбранным значениям:

$$t_u = 2 \text{ c}$$

$$t_3 = 3/4 t_u = 1.5 \text{ c}$$

$$t_u(OCR1A) = t_u(OCR1B) = OCR1A \cdot T_{cnt} = OCR1A \cdot K/F_{clk} = OCR1A \cdot 1024 / (3.69 \cdot 10^6) =$$

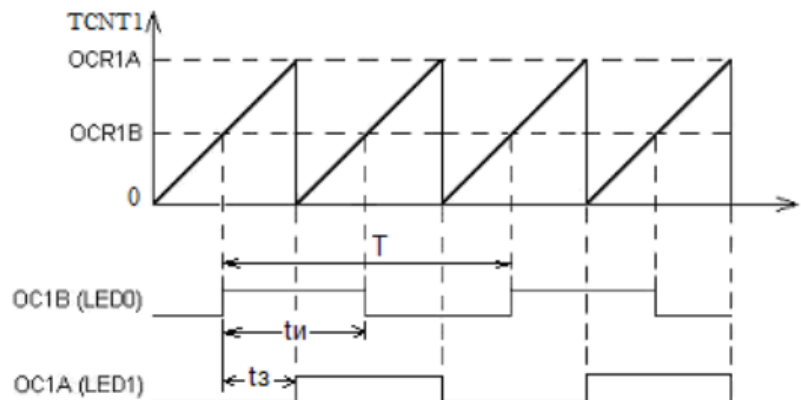
2 c

$$\text{Отсюда } OCR1A = 7207 = 0x1C27$$

$$t_3 = (OCR1A - OCR1B) \cdot K/F_{clk} = (7207 - OCR1B) \cdot 1024 / (3.69 \cdot 10^6) = 1.5 \text{ c}$$

$$\text{Отсюда } OCR1B = 1802 = 0x070A$$

На рисунке 3 изображены временные диаграммы таймера/счетчика T1 при работе в режиме сравнения.



**Рисунок 3 - временные диаграммы таймера/счетчика T1**

Код программы:

```

;*****
;Программа 4.3 для МК ATx8515:
;демонстрация работы функции сравнения таймера T1.
;Частота тактового генератора Fclk=3,69 МГц.
;При нажатии на SW0 (START) запускается счёт с частотой Fclk/K,
;при нажатии на SW2 (STOP) счёт останавливается.
;При совпадении содержимого счётчика и регистра сравнения OCR1B
;переключается светодиод LED0,
;содержимого счётчика и регистра сравнения OCR1A - LED1.
;Соединения: LED0-PE2, LED1-PD5, SW0-PD0, SW2-PD2
;*****
#include "8515def.inc" ;файл определений AT90S8515
#include "m8515def.inc" ;файл определений ATmega8515
.def temp = r16 ;временный регистр
.equ START = 0 ;0-ой вывод порта PD
.org $000

rjmp INIT ;обработка сброса
.org $001

```



rjmp STOP\_PRESSED ;обработка внешнего прерывания INTO -  
; нажатие STOP

\*\*\*Инициализация МК

INIT: ldi temp,low(RAMEND) ;установка  
out SPL,temp ; указателя стека  
ldi temp,high(RAMEND) ; на последнюю  
out SPH,temp ; ячейку ОЗУ  
ldi temp,0x20 ;инициализация вывода PD5  
out DDRD,temp ; как выхода  
ldi temp,0x05 ;включение 'подтягивающих' резисторов  
out PORTD,temp ; в PD0, PD2  
ldi temp,0x04 ;/// для ATmega8515 инициализация вывода порта  
out DDRE,temp ;/// PE2 (OC1B) на вывод  
ldi temp,(1<<INT0) ;разрешение прерывания INTO  
out GICR,temp ; в регистре GICR (или GIMSK)  
clr temp ;обработка прерывания INTO  
out MCUCR,temp ; по низкому уровню

\*\*\*Настройка функции сравнения таймера T1

cli ;запрещение прерываний  
ldi temp,0x50 ;при сравнении состояния выводов OC1A и  
out TCCR1A,temp ; OC1B изменяются на противоположные  
clr temp ;останов  
out TCCR1B,temp ; таймера  
ldi temp,0x07 ;запись числа в  
out OCR1BH,temp ; регистр сравнения,  
ldi temp,0x0A ; первым записывается  
out OCR1BL,temp ; старший байт  
ldi temp,0x1C ;запись числа в  
out OCR1AH,temp ; регистр сравнения,  
ldi temp,0x27 ; первым записывается  
out OCR1AL,temp ; старший байт  
clr temp ;обнуление  
out TCNT1H,temp ; счётного  
out TCNT1L,temp ; регистра  
sei ;разрешение прерываний

WAITSTART: sbic PIND,START ;ожидание нажатия  
rjmp WAITSTART ; кнопки START  
ldi temp,0x0D ;запуск таймера с предделителем K=1024,  
out TCCR1B,temp ; при совпадении с OCR1A - сброс

LOOP: por ; во время цикла происходит  
 rjmp LOOP ; увеличение содержимого счётного регистра

;\*\*\*Обработка прерывания от кнопки STOP

STOP\_PRESSED:

clr temp ;останов

out TCCR1B,temp ; таймера

WAITSTART\_2: ;ожидание

sbic PIND,START ; нажатия

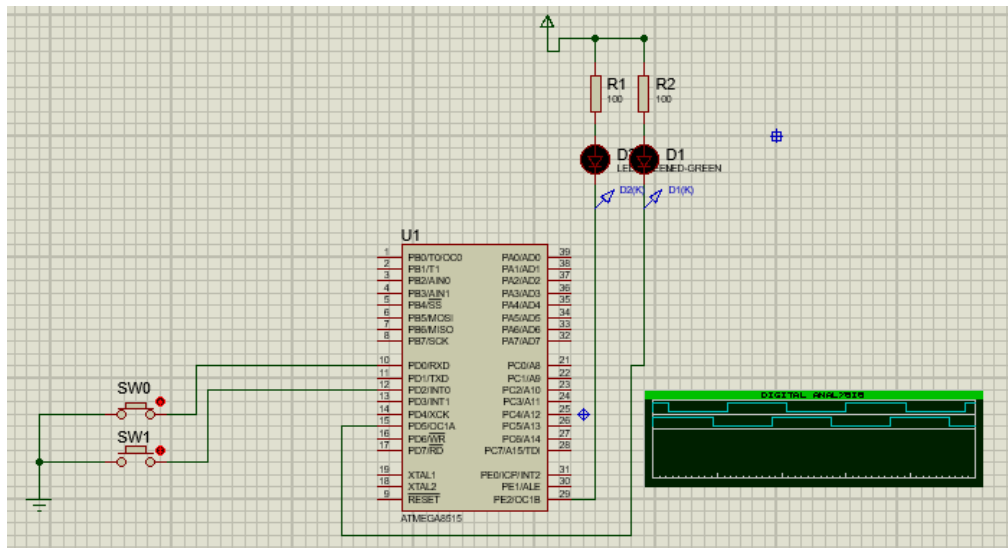
rjmp WAITSTART\_2 ; кнопки START

ldi temp,0x0D ;запуск

out TCCR1B,temp ; таймера с предделителем K=1024

reti

Схема для тестирования программы в Proteus приведена на рисунке 4.



**Рисунок 4 - схема для тестирования программы в Proteus**

Соответствующие временные диаграммы приведены на рисунке 5.  $t_n$  и  $t_z$  несколько меньше расчетных, так как Proteus эмулирует работу микроконтроллера с частотой 4 МГц, вместо расчетных 3.69 МГц.

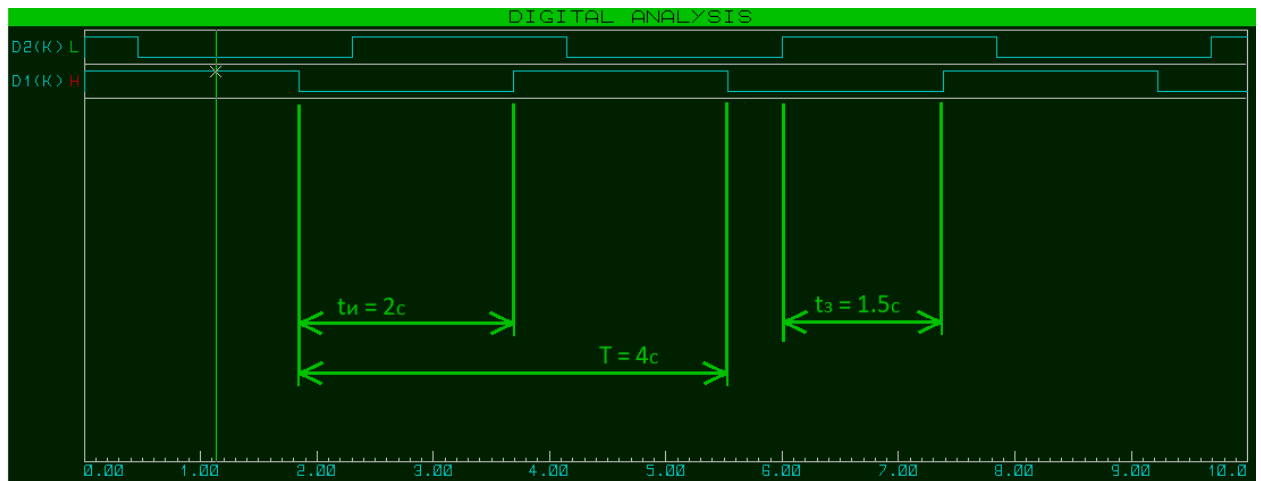


Рисунок 5 – временные диаграммы в режиме сравнения

#### Задание 4. Режим ШИМ

Проверить работу программы. Изменить программу для формирования двух выходных ШИМ-сигналов с коэффициентом заполнения  $\frac{1}{4}$  или  $\frac{3}{4}$  для 10-разрядного режима ШИМ. Формирование ШИМ сигналов представлено на рисунке 6.

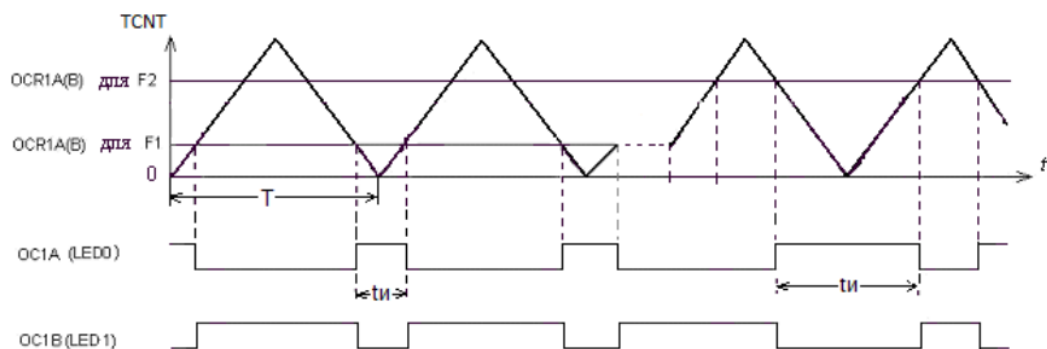


Рисунок 6 - Формирование ШИМ-сигналов OC1A (неинвертированный) и OC1B (инвертированный)

ШИМ 10-разрядный, поэтому модуль счёта TOP=1023.  $K = 1024$ .

$$T = 2 * TOP * K / F_{ck} = 2 * 1023 * 1024 / (3,69 * 10^6) \approx 0,57c$$

$$t_H = 3/4 T \approx 0,43 c$$

$$t_L = 1/4 T \approx 0,14 c$$

Тогда для F1:

$$OCR1A = OCR1B = (1/4) * TOP \approx 256 = 0x100$$

Для F2:

$$OCR1A = (3/4) * TOP \approx 767 = 0x2FF$$

Код измененной программы:

```
;*****  
;  
*  
;Программа 4.4 для МК АТх8515:  
;демонстрация работы таймера T1 в режиме ШИМ  
;При нажатии SW0 происходит генерация  
;ШИМ-сигналов с порогом сравнения F1  
;При нажатии SW1 происходит генерация  
;ШИМ-сигналов с порогом сравнения F2  
;Соединения: PD5-LED0,PE2-LED1, PD0-SW0,PD1-SW1  
;*****  
;  
*  
;include "8515def.inc" ;файл определений AT90S8515  
.include "m8515def.inc" ;файл определений ATmega8515  
.def temp = r16 ;временный регистр  
;***Выводы порта PD  
.equ SW0 = 0  
.equ SW1 = 1  
.org $000  
rjmp INIT ;обработка сброса  
  
;***Инициализация МК  
INIT: ldi temp,0x20 ;инициализация PD5  
out DDRD,temp ; на вывод  
ldi temp,0x03 ;включение 'подтягивающих'  
out PORTD,temp ; резисторов порта PD  
ldi temp,0x04 ;/// для ATmega8515 инициализация  
out DDRE,temp ;/// PE2 (OC1B) на вывод  
  
cli ;запрещение прерываний  
;настройка таймера: 10-разрядный режим ШИМ, на выводе  
;OC1A неинвертированный сигнал, OC1B – инвертированный сигнал  
ldi temp,0xB3  
out TCCR1A,temp  
clr temp ; обнуление  
out TCNT1H,temp ; счётного  
out TCNT1L,temp ; регистра  
ldi temp,0x05 ;таймер  
out TCCR1B,temp ; запущен с предделителем 1024  
F1: sbic PIND,SW0 ;проверка нажатия SW0  
rjmp F2
```

```

;***Установка порога F1
ldi temp,0x01 ;запись числа в
out OCR1AH,temp ; регистры сравнения,
out OCR1BH,temp ; первым записывается
ldi temp,0x00 ; старший байт
out OCR1AL,temp
out OCR1BL,temp
F2: sbic PIND,SW1 ; проверка нажатия SW1
rjmp F1

```

```

;*** Установка порога F2
ldi temp,0x02 ;запись числа в
out OCR1AH,temp ; регистры сравнения,
out OCR1BH,temp ; первым записывается
ldi temp,0xff ; старший байт
out OCR1AL,temp
out OCR1BL,temp
rjmp F1

```

Схема для тестирования программы в Proteus представлена на рисунке

7.

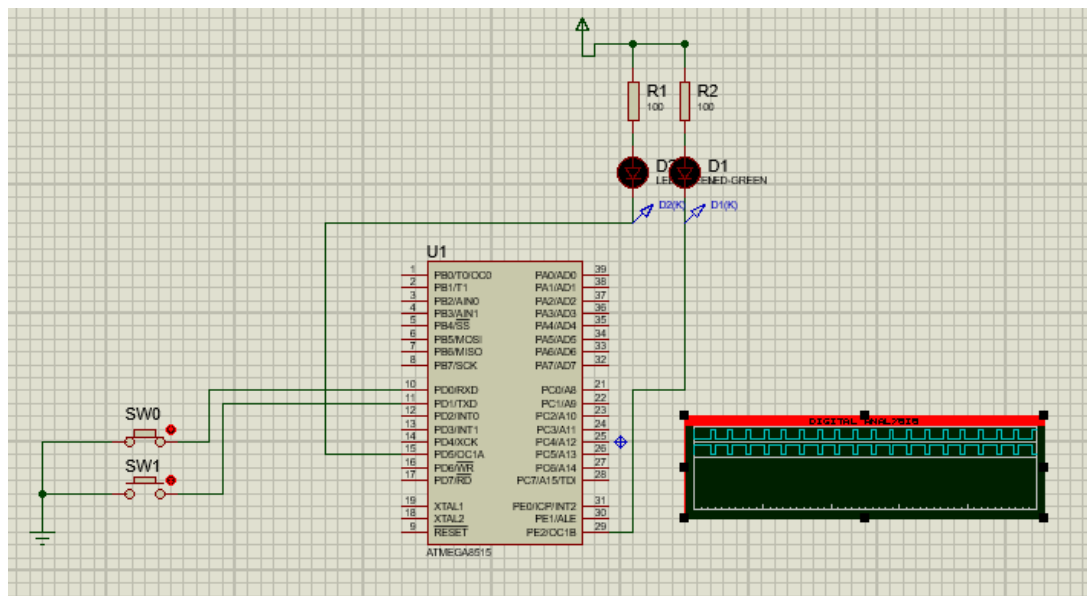


Рисунок 7 - схема для тестирования программы в Proteus

Соответствующая временная диаграмма представлена на рисунке 8.

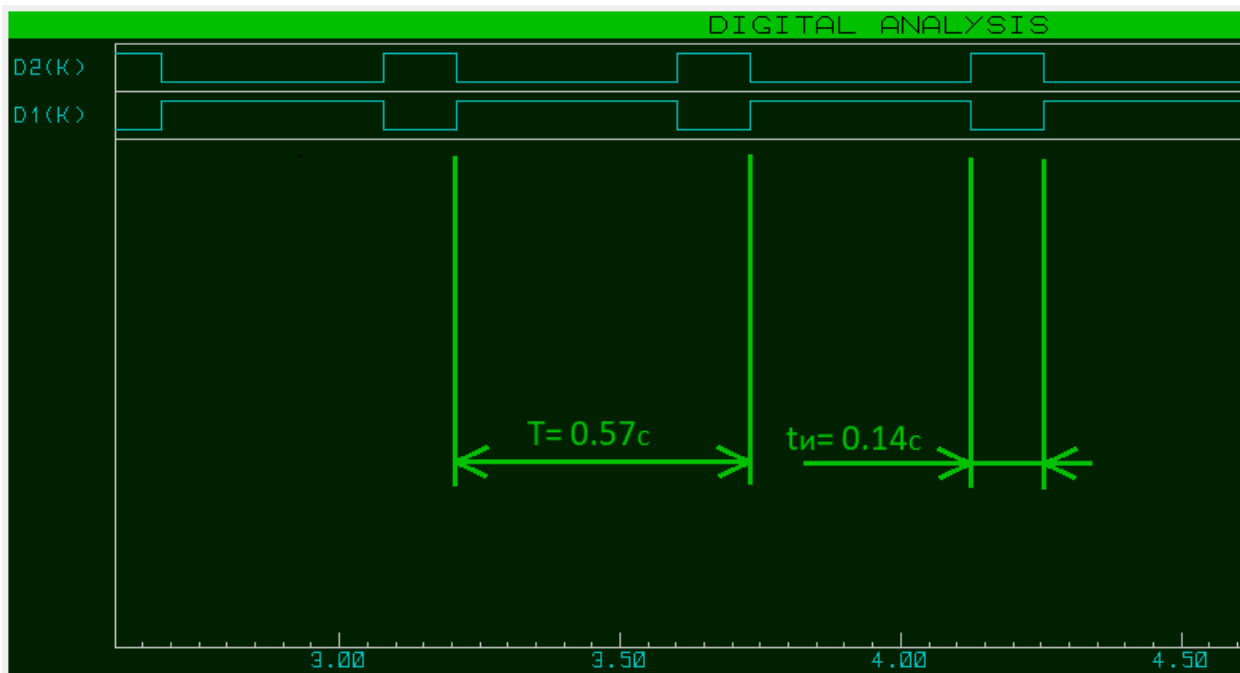


Рисунок 8 – временные диаграммы в режиме ШИМ

### Задание 5

Загрузить программу в STK500. Для проверки работы программы включите одновременно секундомер часов и запустите программу. После останова программы проверьте показания времени на часах и в регистрах захвата таймера. Результаты представлены в таблице 1.

Таблица 1 — результаты измерений

Время на секундомере, с	Значения в регистрах захвата таймера	Расчетное время, с
5.51	0100 1100 0010 0111 (19495)	$19495 \cdot (1024 / (3.69 \cdot 10^6)) \approx 5.41$
8.32	0111 0100 1011 0001 (29873)	$29873 \cdot (1024 / (3.69 \cdot 10^6)) \approx 8.29$
3.15	0010 1101 0000 1011 (11531)	$11531 \cdot (1024 / (3.69 \cdot 10^6)) \approx 3.20$

**Вывод:** в ходе данной лабораторной работы было изучено устройство таймеров T0 и T1 в контроллерах AVR. Были изучены следующие режимы таймеров:

- режим счетчика (вызывается прерывание при переполнении счетчика событий);
- режим таймера (вызывается прерывание при переполнении таймера);
- режим сравнения (вызывается прерывание при совпадении значения таймера со значением одного из регистров сравнения);
- режим ШИМ (счётчик TCNT1 функционирует как реверсивный счётчик, используется схема сравнения);
- режим захвата (функция заключается в сохранении в определённый момент времени состояния таймера/счётчика TCNT1 в регистре захвата ICR1);

Во всех режимах, кроме режима счетчика, используется коэффициент делителя, позволяющий изменить частоту работы таймера (поделить частоту тактового сигнала микроконтроллера на 1, 8, 64, 256, 1024).