

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**КАФЕДРА САУ**

**ОТЧЕТ**  
**по Лабораторной работе №5**  
**по дисциплине «МИКРОПРОЦЕССОРНАЯ ТЕХНИКА В**  
**МЕХАТРОНИКЕ**  
**И РОБОТОТЕХНИКЕ»**  
**тема: Работа с ультразвуковым дальномером**

Студенты гр. 6492

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Мурашко А.С.  
Огурецкий Д.В.  
Спорыш И.В.

Преподаватель

\_\_\_\_\_

Девяткин А.В.

Санкт-Петербург

2019

**Цель работы** — Освоение работы с ультразвуковым датчиком.

**Задание на лабораторную работу.** Написать программу вывода расстояния от ультразвукового датчика в сантиметрах на семисегментные индикаторы.

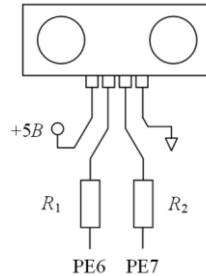


Рис. 1 Схема включения ультразвукового датчика расстояния

## Выполнение работы

### Описание прерывания

Данный алгоритм производит сохранение ширины импульса.

#### Код

```
1 interrupt [EXT_INT7] void extIntIsr(void)
2 {
3     // Если высокий уровень на ножке, то
4     // это начало импульса
5     if(BIT_IS_SET(PINE, 7))
6     {
7         // Обнуляем счётный регистр таймера №3
8         TCNT3H = 0;
9         TCNT3L = 0;
10    }
11    else
12    {
13        // Если низкий уровень на ножке, то
14        // это конец импульса
15        // Сохраняем текущее значение счётного регистра
16        // таймера №3
17        impulseWidth = TCNT3L;           //tick
18        impulseWidth += (TCNT3H) << 8; //tick
19    }
20 }
```

### Описание main()

Работа программы подробно описана в блок-схеме.

## Код

```

1  #define prop  3/250  //0.012 cm/tick
2
3  void main(void)
4  {
5      unsigned int length;
6
7      init_segments();
8      DDRE = _BV(DDE6); // PORTE6 для подачи импульса на вход датчика
9
10     EICRB = _BV(ISC70) ; //настройка внешнего прерывания на любое измене-
11     ние входного сигнала
12     EIMSK = _BV(INT7); //разрешение прерывания
13     #asm("sei"); //разрешение прерываний
14
15     // Инициализация таймера №3
16     TCCR3B = _BV(CS31); //freq=11.0529/8=1.382 MHz T=724 ns/tick
17
18     while(1)
19     {
20         length =impulseWidth*prop; //cm   преобразование
21         BitSet(PORTE,DDE6);
22         delay_us(10);
23         BitClr(PORTE,DDE6);
24         delay_ms(50); //пауза для ожидания окончания измерения
25         indic_uint(length); //вывод значения на
26     }

```

## Блок-схема

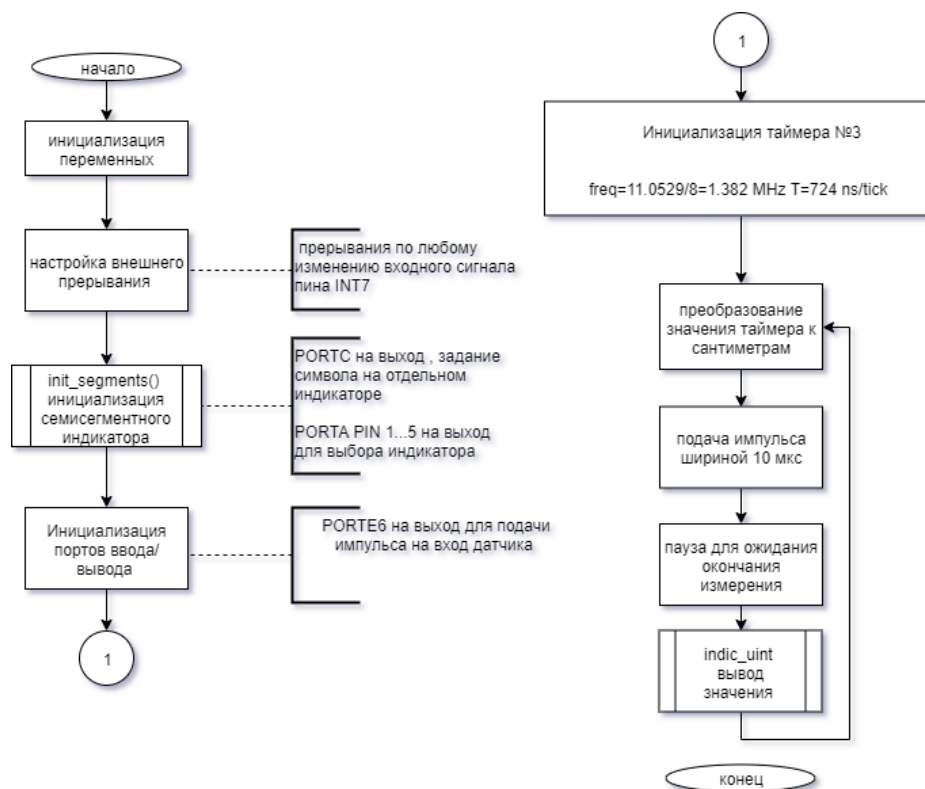


Рис. 5 Блок-схема main()

Вывод: с помощью таймера-счетчика можно управлять измерительными датчиками и преобразовывать результат в тот вид, который требуется для дальнейшей обработки.