

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

ОТЧЕТ
по Лабораторной работе №4
по дисциплине «МИКРОПРОЦЕССОРНАЯ ТЕХНИКА В
МЕХАТРОНИКЕ
И РОБОТОТЕХНИКЕ»
тема: Работа с сервомашинками

Студенты гр. 6492

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

Преподаватель

Девяткин А.В.

Санкт-Петербург

2019

Цель работы — Освоение работы с сервомашинками.

Задание на лабораторную работу. Написать программу управления сервомашинкой с помощью ползункового потенциометра.

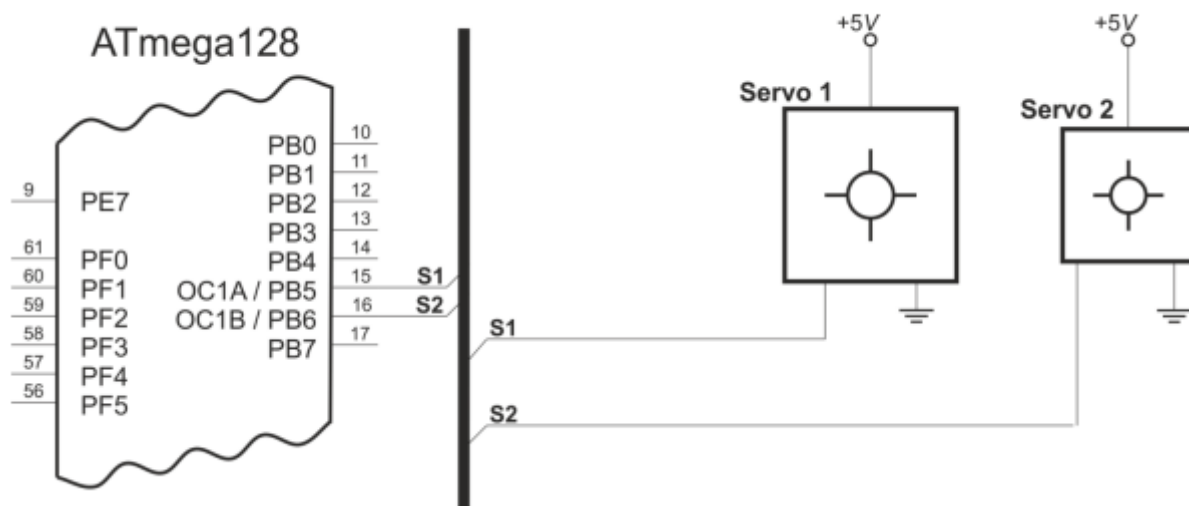


Рис. 1 Схема подключения сервомашинок

Выполнение работы

Описание функции измерения напряжения с помощью АЦП

Данный алгоритм уже приведен в лабораторной работе №3.

Описание main()

В данной программе мы используем оба байта (младший и старший) регистра сравнения OCR, но максимальное значение, которое может быть записано в него это 1023, так как АЦП имеет 10-ти битную разрядность, поэтому перед записью его в регистры OCR необходимо преобразовать его, чтобы он лежал в интервале значений MIN_POS и MAX_POS. Для этого мы домножаем его на константу, которая равна $\frac{MAX_POS - MIN_POS}{1023} = 1,351$.

Код

```
1 // Макроопределение значения регистра ICR1
2 #define ICR_VALUE 1382400 / 50 //27648
3 // Макроопределения значений регистров OCR1A и OCR1B
4 // Соответствующих крайним положениям
5 #define MAX_POS ICR_VALUE / 10 //2764.8
6 #define MIN_POS ICR_VALUE / 20 //1382.4
7 #define PROPORTIONAL (MAX_POS-MIN_POS)/1023 //1.351
8 void main(void)
9 {
10     float val;
11     init_segments();
12     ADMUX = _BV(REFS0) | 3; // Выбор канала с потенциометром
13     ADCSRA |= _BV(ADEN); //инициализация АЦП
14
15     // Инициализация портов ввода/вывода для управления сервомашинками
16     // PORTB5 (OC1A) – Servo 1
17     // PORTB6 (OC1B) – Servo 2
18     DDRB |= _BV(5) | _BV(6);
19     // Инициализация таймера №1
20     //Сброс OC1A при совпадении значения TCNT3 с OCR3A (установка 0)
21     //Сброс OC1B при совпадении значения TCNT3 с OCR3B (установка 0)
22     TCCR1A = _BV(COM1A1) | _BV(COM1B1) | _BV(WGM11);
23     // быстрая ШИМ, верхний предел ICRn
24     // Ftimer = Fcpu / 8
25     TCCR1B = _BV(WGM12) | _BV(WGM13) | _BV(CS11);
26     // Установка верхнего предела для таймера №1
27     ICR1 = ICR_VALUE;
28     OCR1A=MIN_POS;
29     OCR1B=MIN_POS;
30
31     while(1)
32     {
33         val=readAdc();
34         // преобразование значения к диапазону регулирования
35         val*=1.351;
36         val+=MIN_POS;
37
38         indic_int(val);
39         // Задание положения сервомашинки
```

```

40     OCR1A=(int) val;
41     OCR1B=(int) val;
42     delay_ms(20);
43 }
44 }

```

Блок-схема

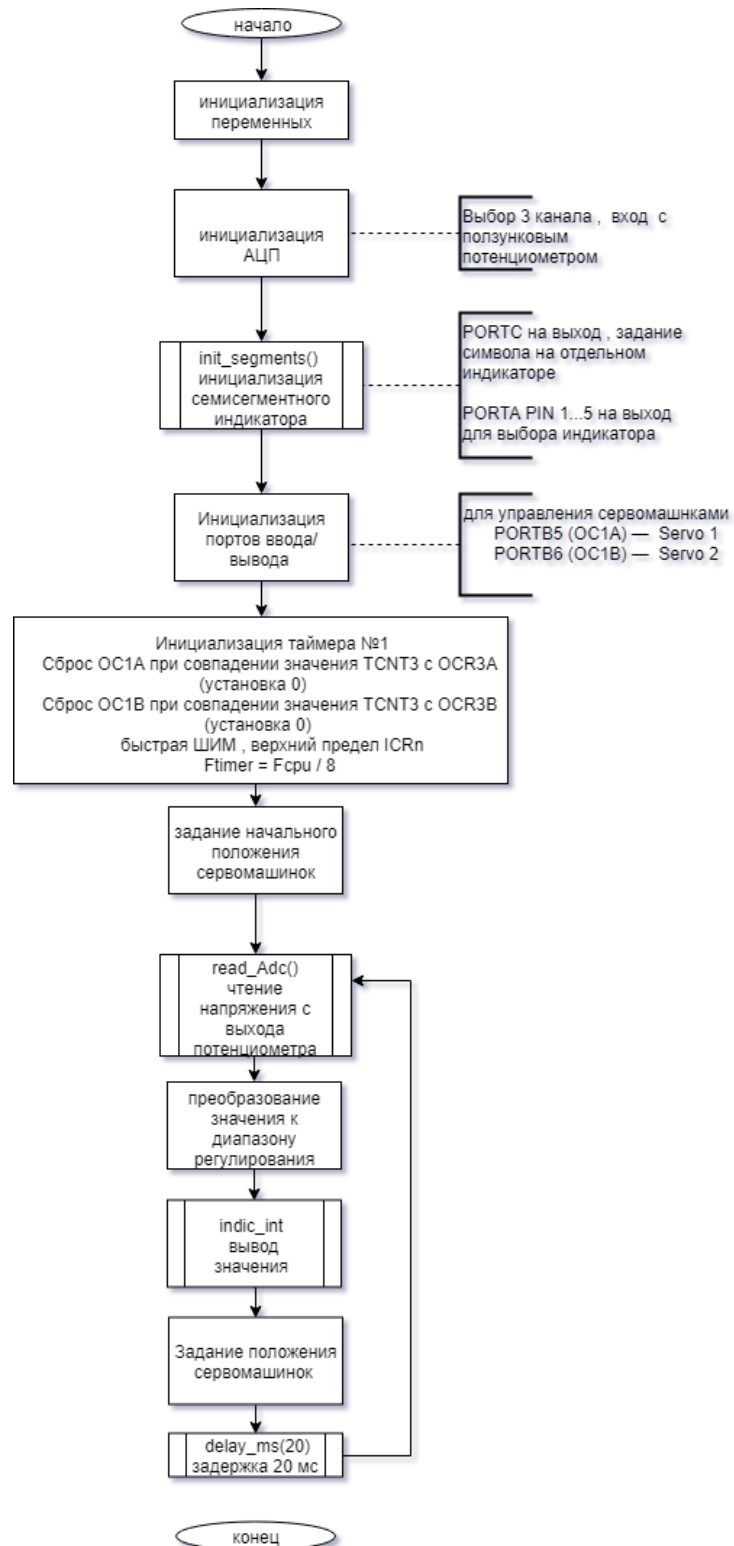


Рис. 5 Блок-схема main()

Вывод: с помощью АЦП и таймера-счетчика с использованием PDM (Pulse Duration Modulation) можно задавать импульсы для схемы управления сервопривода, которая изменяет угол поворота выходного вала сервопривода. Здесь используется именно способ PDM, так как крайне важна длина импульсов и не так важна частота их появления.