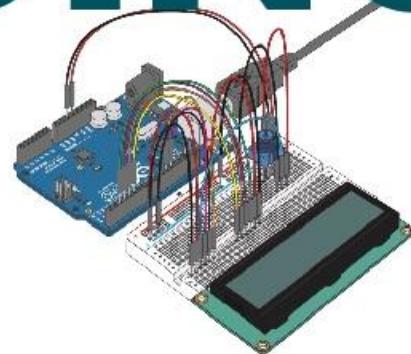
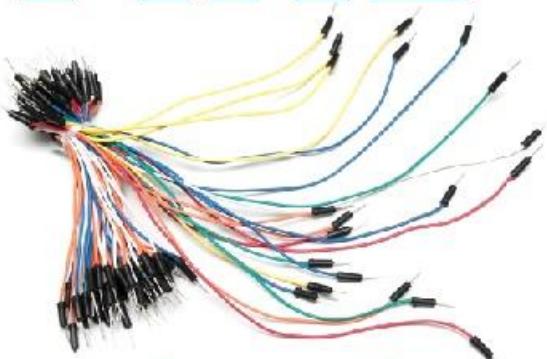


# ARDUINO



**УМНАЯ МЕХАНИКА**

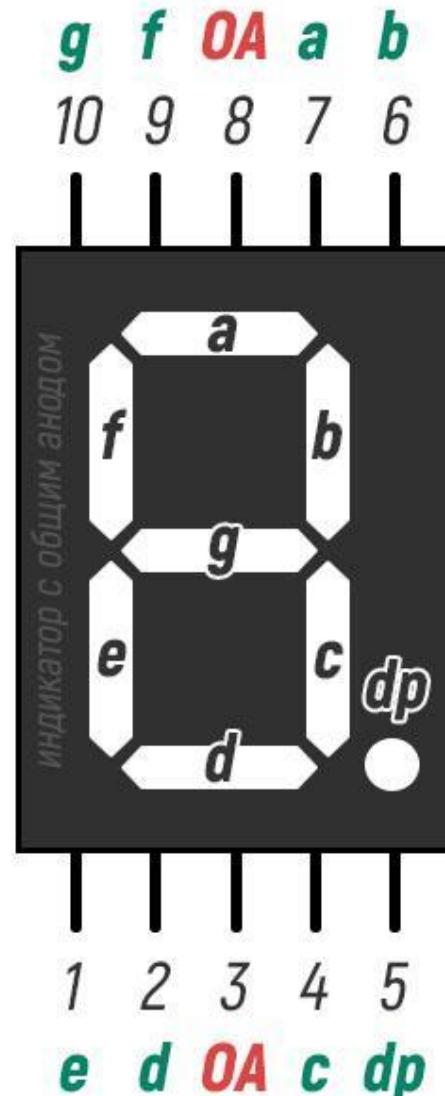
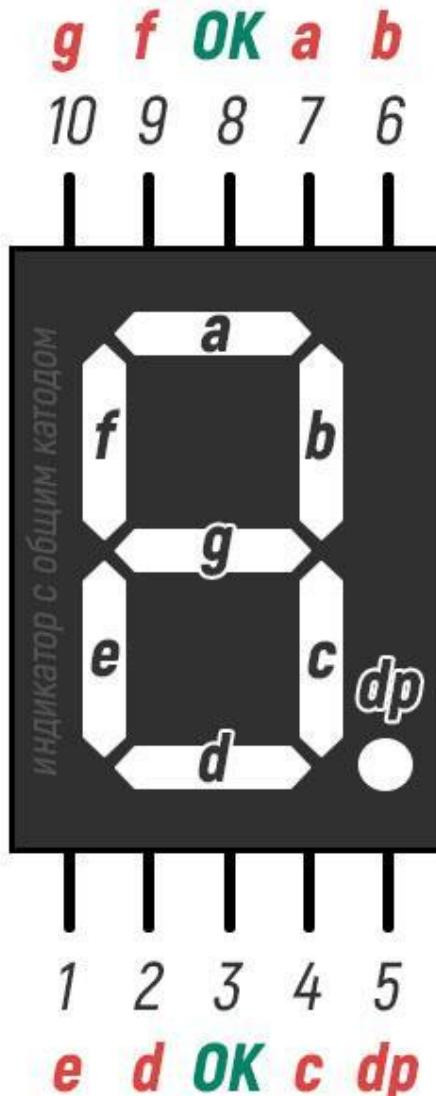
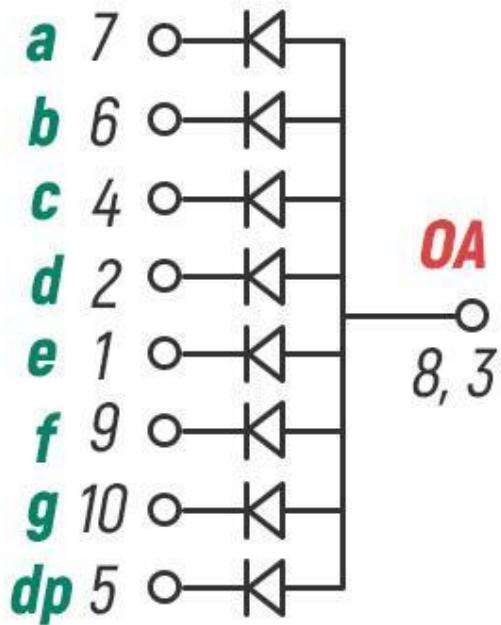
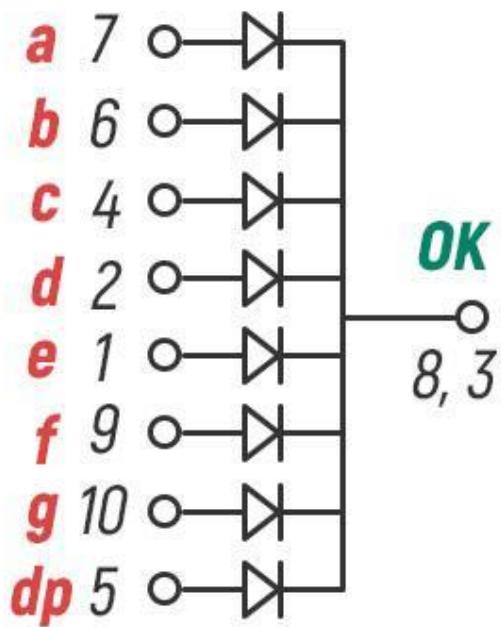


1

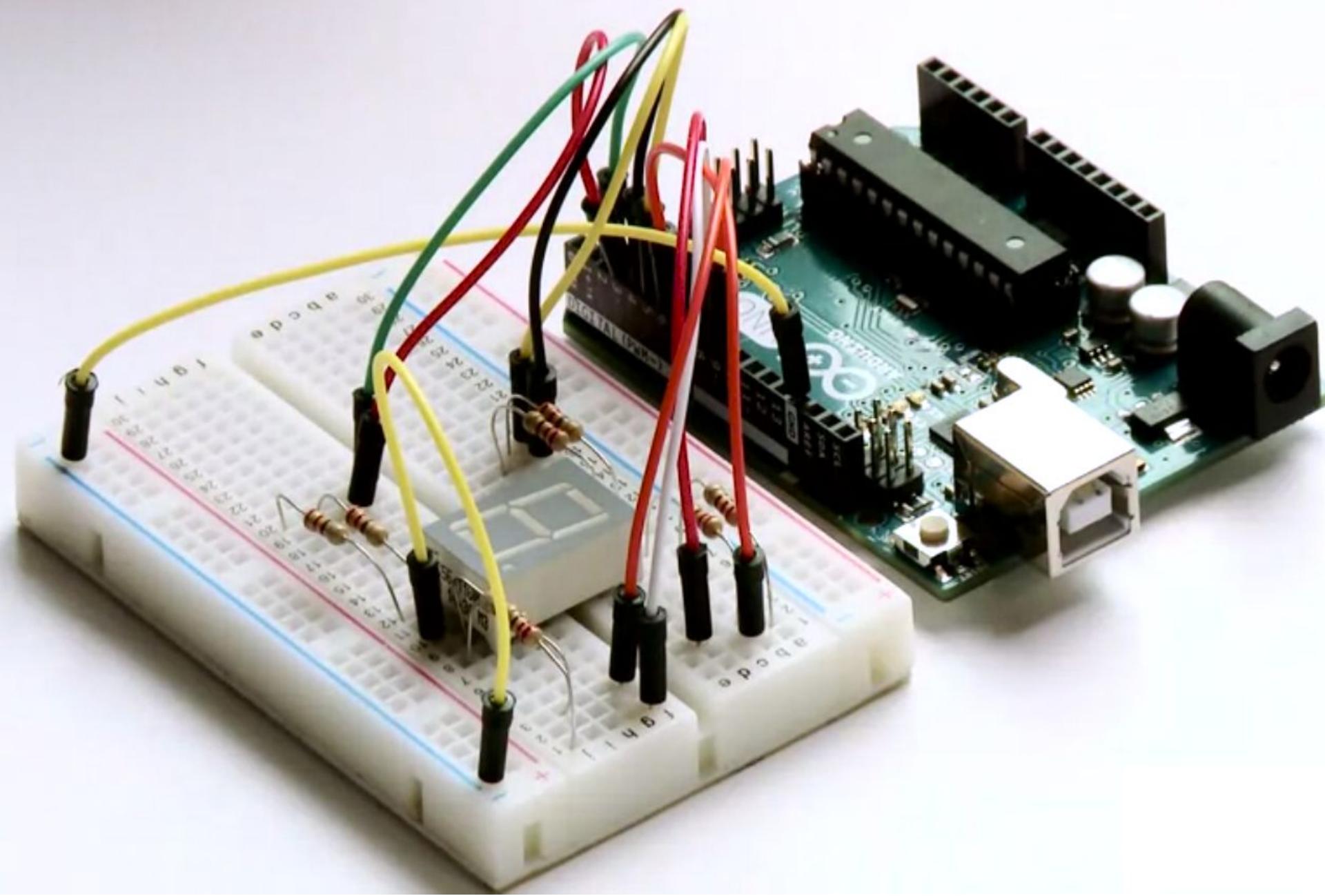
1

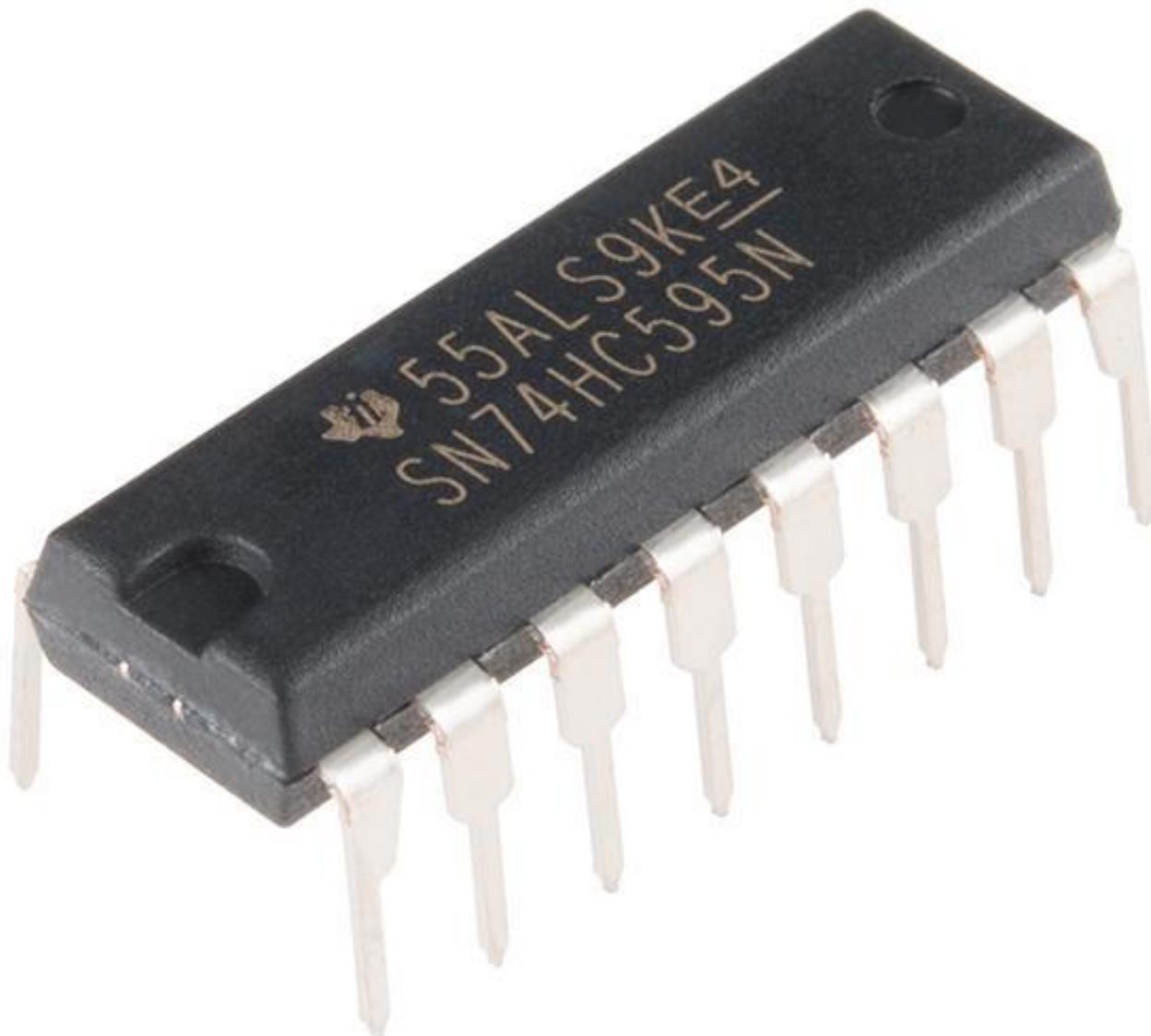
R

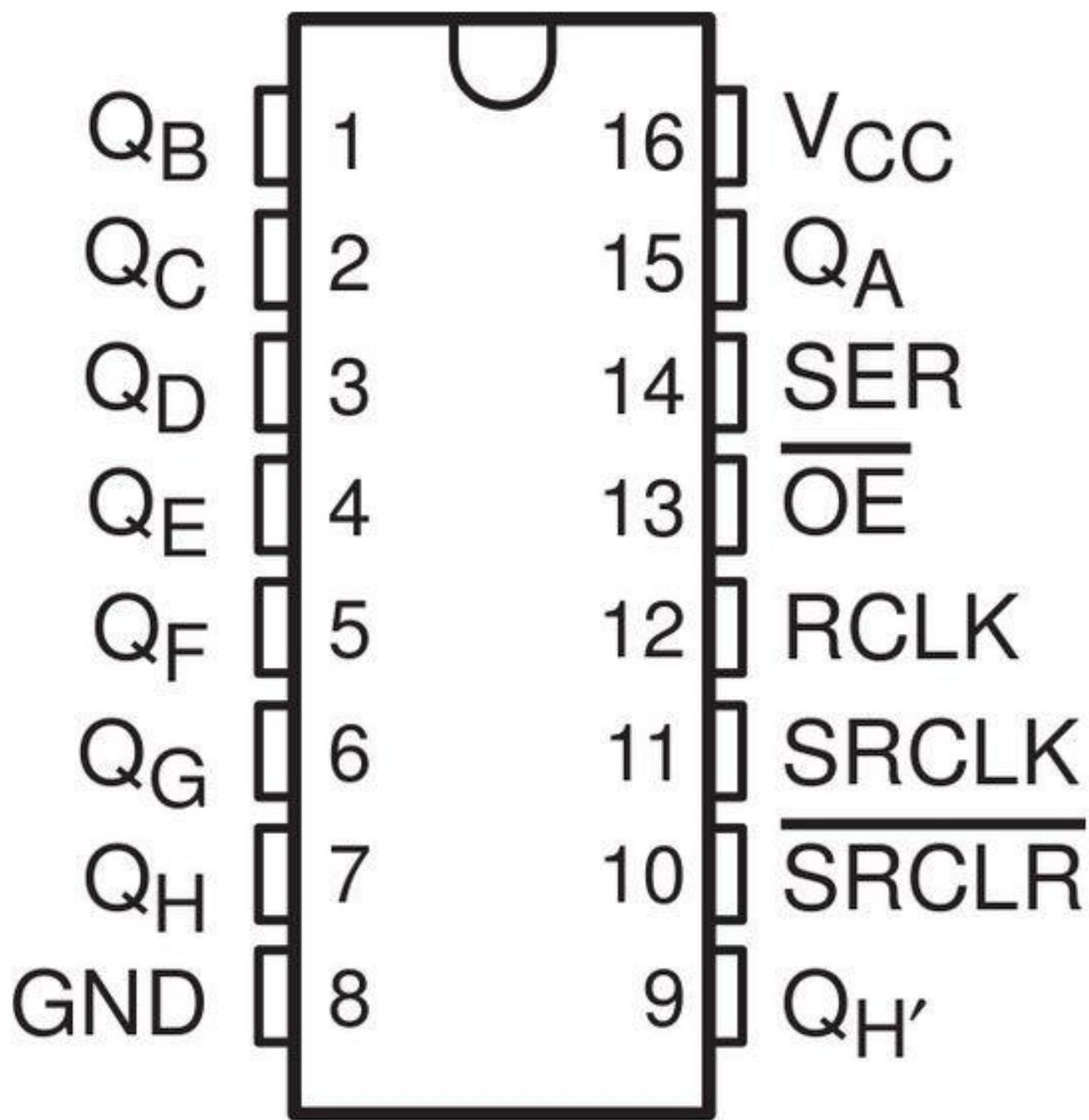
266.933RH

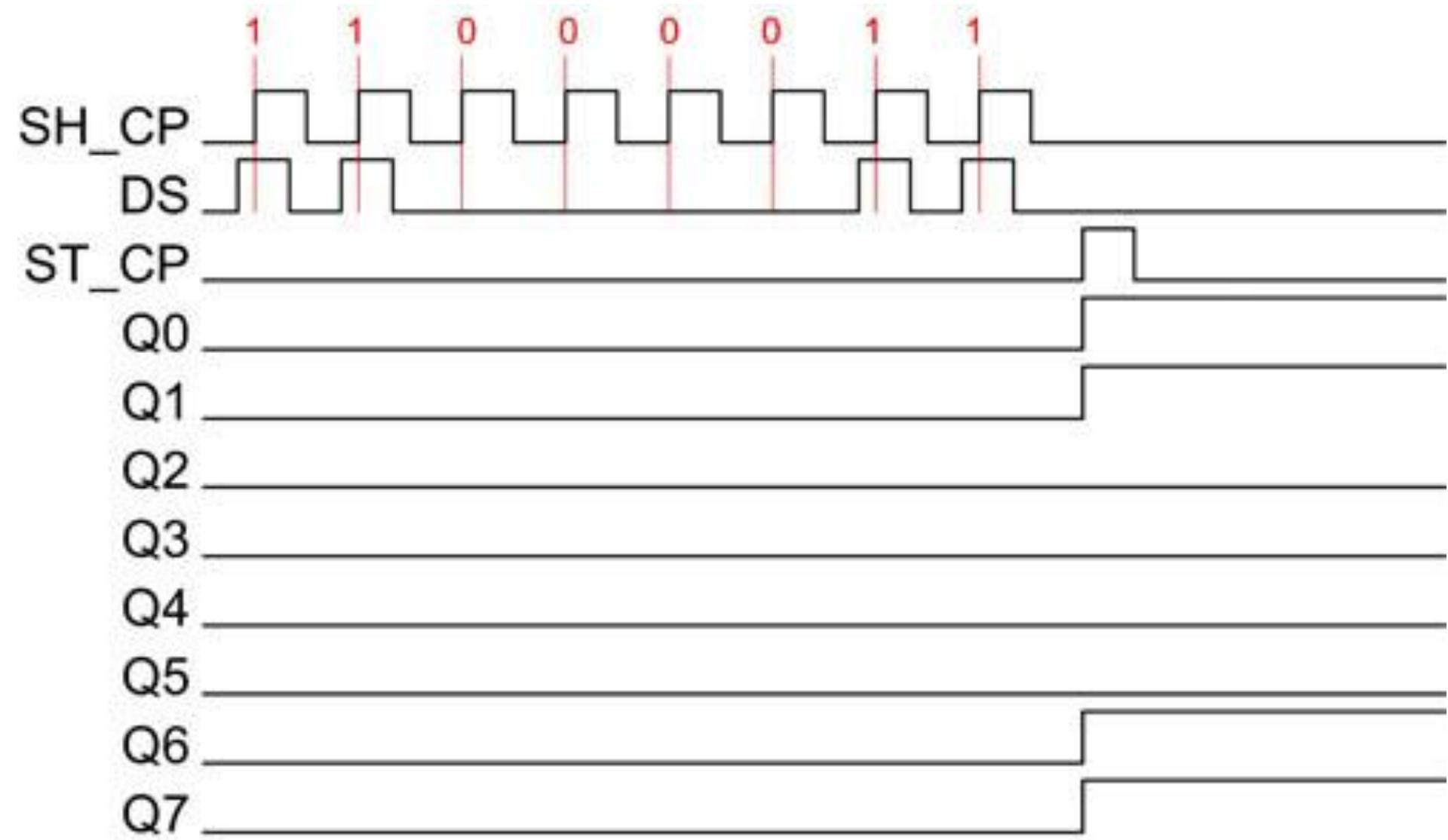


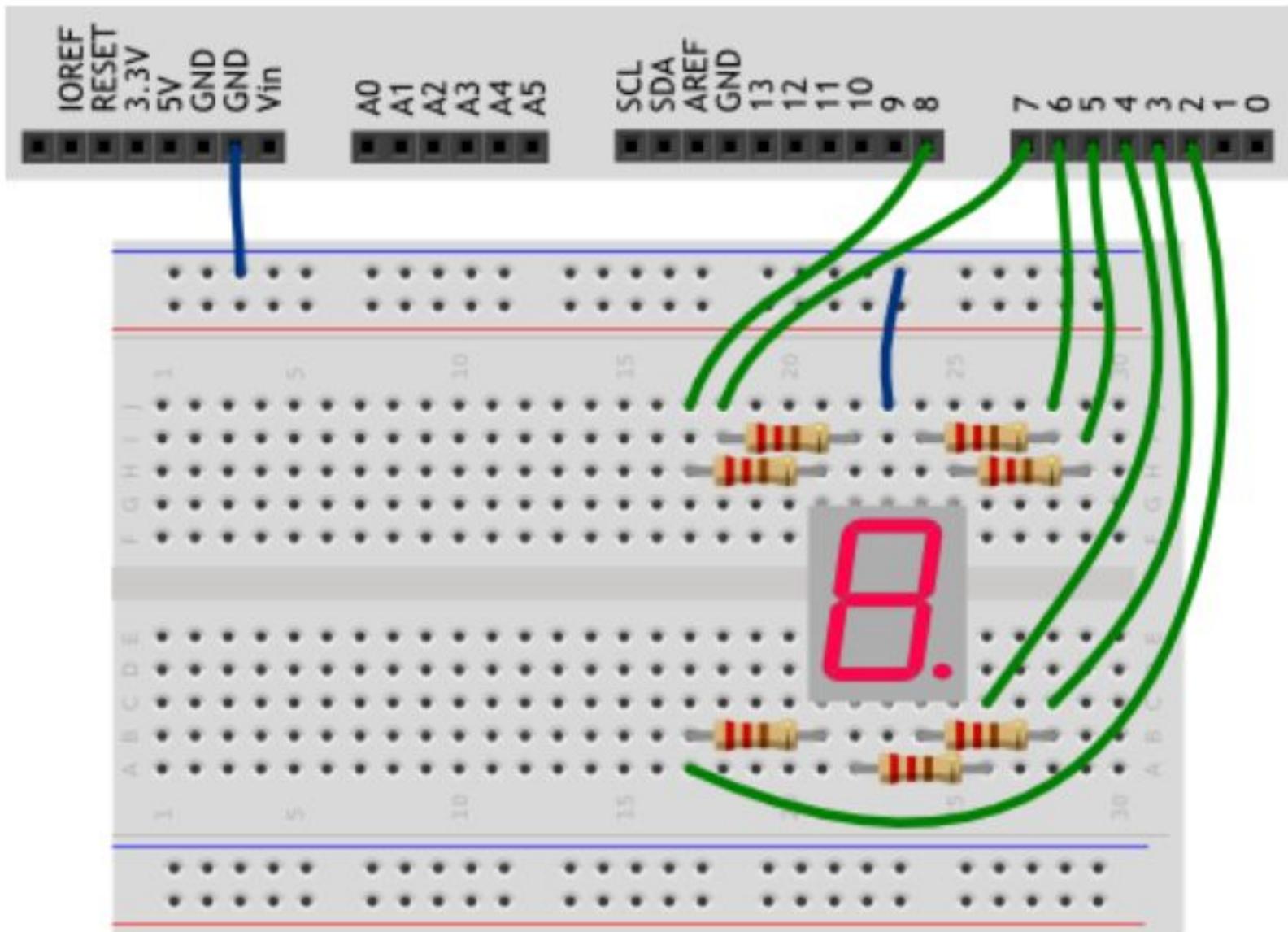
**OK – общий катод** сегменты индикатора:  
**OA – общий анод**











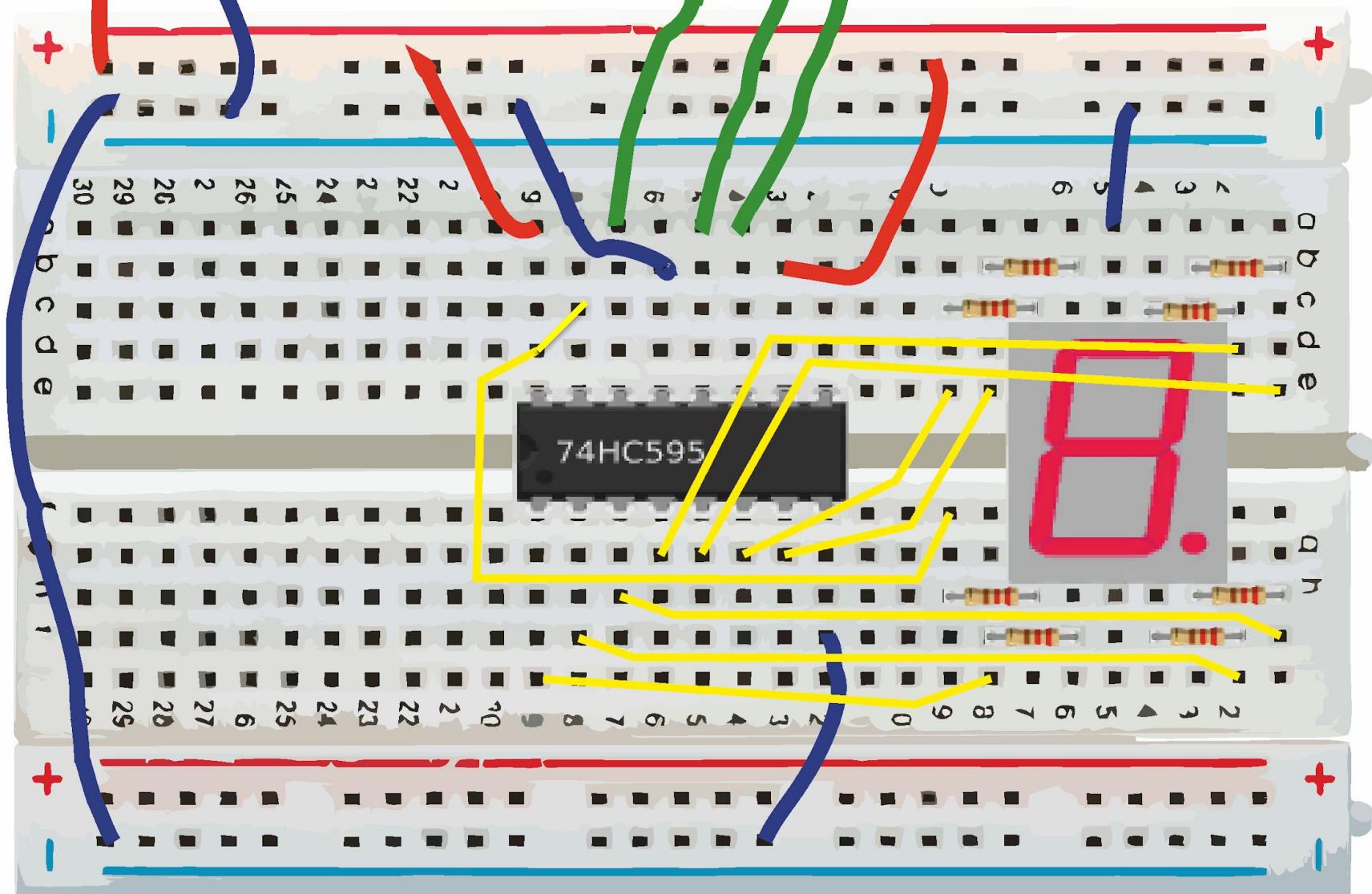
```
#define FIRST_SEGMENT_PIN    2
#define SEGMENT_COUNT          7

// префикс «0b» означает, что целое число за ним записано в
// в двоичном коде. Единицами мы обозначим номера сегментов
// индикатора, которые должны быть включены для отображения
// арабской цифры. Всего цифр 10, поэтому в массиве 10 чисел.
// Нам достаточно всего байта (англ. byte, 8 бит) для хранения
// комбинации сегментов для каждой из цифр.

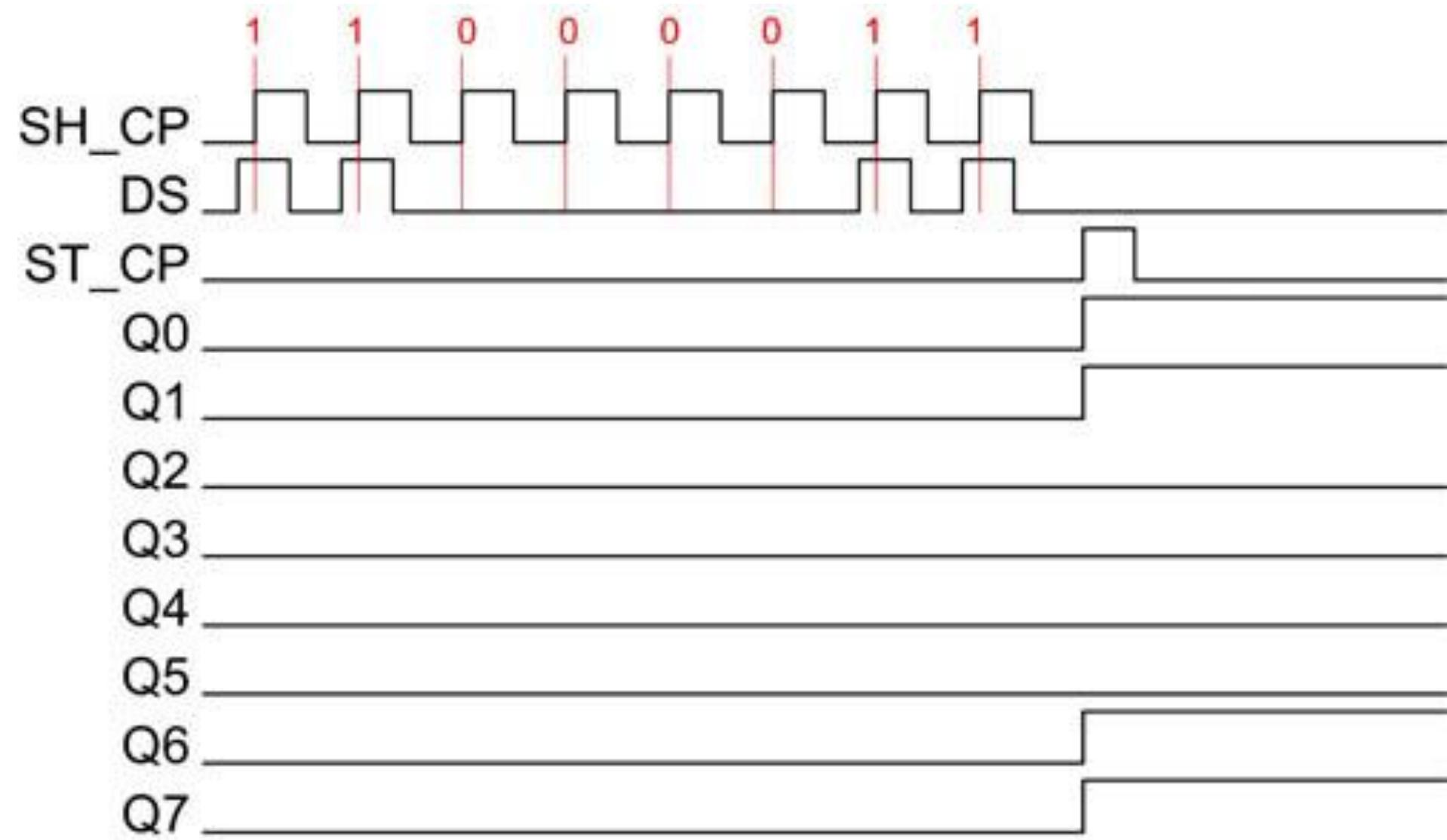
byte numberSegments[10] = {
    0b000111111, 0b00001010, 0b01011101, 0b01011110, 0b01101010,
    0b01110110, 0b01110111, 0b00011010, 0b01111111, 0b01111110,
};

void setup()
{
    for (int i = 0; i < SEGMENT_COUNT; ++i)
        pinMode(i + FIRST_SEGMENT_PIN, OUTPUT);
}
```

```
void loop()
{
    // определяем число, которое собираемся отображать. Пусть им
    // будет номер текущей секунды, зацикленный на десятке
    int number = (millis() / 1000) % 10;
    // получаем код, в котором зашифрована арабская цифра
    int mask = numberSegments[number];
    // для каждого из 7 сегментов индикатора...
    for (int i = 0; i < SEGMENT_COUNT; ++i) {
        // ...определяем: должен ли он быть включён. Для этого
        // считываем бит (англ. read bit), соответствующий текущему
        // сегменту «i». Истина — он установлен (1), ложь — нет (0)
        boolean enableSegment = bitRead(mask, i);
        // включаем/выключаем сегмент на основе полученного значения
        digitalWrite(i + FIRST_SEGMENT_PIN, enableSegment);
    }
}
```



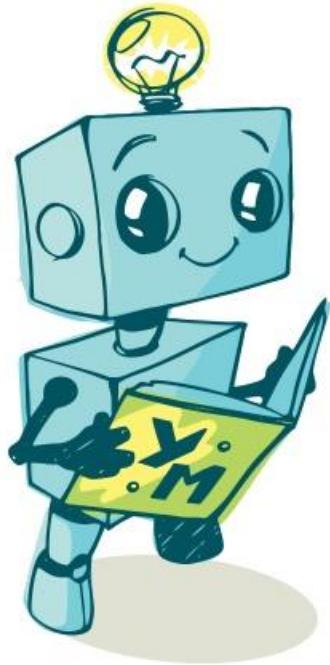
```
1 #define DATA_PIN 13
2 #define LATCH_PIN 12
3 #define CLOCK_PIN 11
4 byte d0 = 0b11101110;
5 byte d1 = 0b00101000;
6 byte d2 = 0b11001101;
7 byte d3 = 0b01101101;
8 byte d4 = 0b00101011;
9 byte d5 = 0b01100111;
10 byte d6 = 0b11100111;
11 byte d7 = 0b00101100;
12 byte d8 = 0b11101111;
13 byte d9 = 0b01101111;
14 void setup() {
```



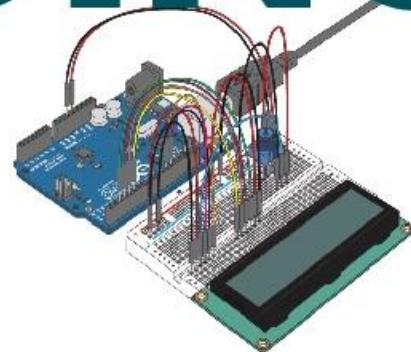
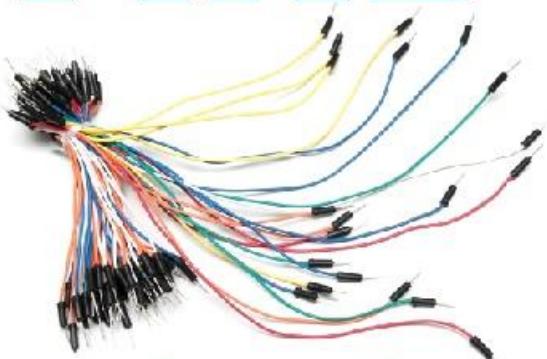
```
1 #define DATA_PIN 13
2 #define LATCH_PIN 12
3 #define CLOCK_PIN 11
4 byte d0 = 0b11101110;
5 byte d1 = 0b00101000;
6 byte d2 = 0b11001101;
7 byte d3 = 0b01101101;
8 byte d4 = 0b00101011;
9 byte d5 = 0b01100111;
10 byte d6 = 0b11100111;
11 byte d7 = 0b00101100;
12 byte d8 = 0b11101111;
13 byte d9 = 0b01101111;
14 void setup() {
```

```
15 void setup() {  
16     pinMode(DATA_PIN, OUTPUT);  
17     pinMode(LATCH_PIN, OUTPUT);  
18     pinMode(CLOCK_PIN, OUTPUT);  
19     Serial.begin(9600); }  
20 void loop() {  
21     if(Serial.available() > 0)  
22     { char digit = Serial.read();  
23  
24         if( digit == '0')  
25         {digitalWrite(LATCH_PIN, LOW);  
26             shiftOut(DATA_PIN, CLOCK_PIN, LSBFIRST, d0);  
27             digitalWrite(LATCH_PIN, HIGH);  
28         }  
29         else if( digit == '1')  
30         {digitalWrite(LATCH_PIN, LOW);
```

```
20 void loop() {  
21   if(Serial.available() > 0)  
22   { char digit = Serial.read();  
23  
24     if( digit == '0')  
25     {digitalWrite(LATCH_PIN, LOW);  
26      shiftOut(DATA_PIN, CLOCK_PIN, LSBFIRST, d0);  
27      digitalWrite(LATCH_PIN, HIGH);  
28    }  
29    else if( digit == '1')  
30    {digitalWrite(LATCH_PIN, LOW);  
31      shiftOut(DATA_PIN, CLOCK_PIN, LSBFIRST, d1);  
32      digitalWrite(LATCH_PIN, HIGH);  
33    }  
34    else if(digit == '2')  
35    {digitalWrite(LATCH_PIN, LOW);  
36      shiftOut(DATA_PIN, CLOCK_PIN, LSBFIRST, d2);  
37      digitalWrite(LATCH_PIN, HIGH); }  
38 }
```



# ARDUINO



**УМНАЯ МЕХАНИКА**