



# Arduino

Четырехразрядный 7-ми  
сегментный индикатор

# Разминка

- Как подключается 7-ми сегментный индикатор?
- Перечислите выводы индикатора?
- Что можно вывести на данном индикаторе?

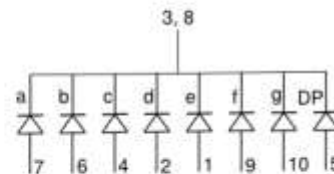
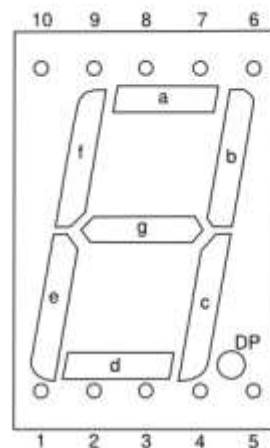
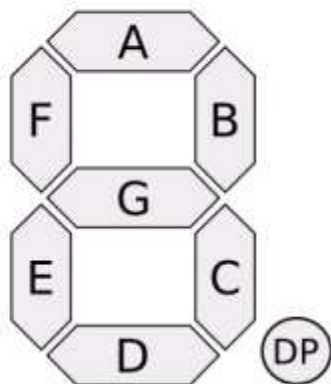
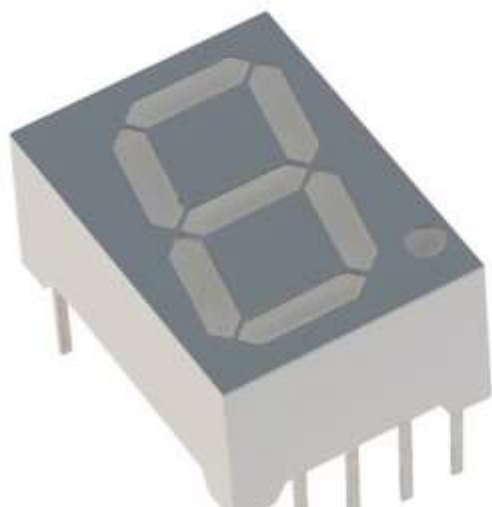
## 7-ми сегментный индикатор

Семисегментный светодиодный индикатор — устройство отображения цифровой информации. Это — наиболее простая реализация индикатора, который может отображать арабские цифры. Для отображения букв используются более сложные многосегментные и матричные индикаторы.

Семисегментные светодиодные индикаторы бывают разных цветов, обычно это белый, красный, зеленый, желтый и голубой цвета. Кроме того, они могут быть разных размеров.

# Устройство индикатора

Семисегментный светодиодный индикатор, как говорит его название, состоит из семи элементов индикации (сегментов), включающихся и выключающихся по отдельности. Включая их в разных комбинациях, из них можно составить упрощённые изображения арабских цифр. Сегменты обозначаются буквами от А до G; восьмой сегмент — десятичная точка (decimal point, DP), предназначенная для отображения дробных чисел. Изредка на семисегментном индикаторе отображают буквы.

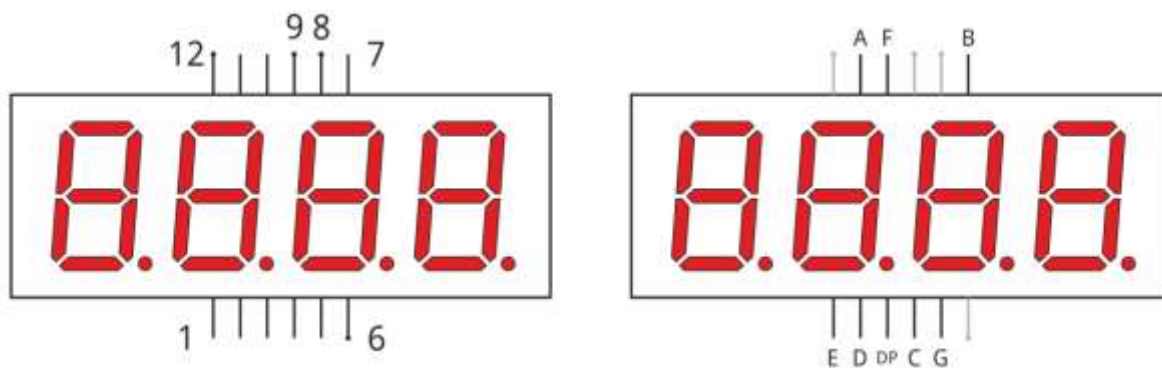


# Отображение на индикаторе

Всего семисегментный светодиодный индикатор может отобразить 128 символов.



Нумерация контактов стандартная — из нижнего левого угла против часовой стрелки:



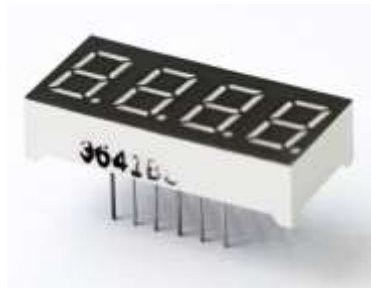
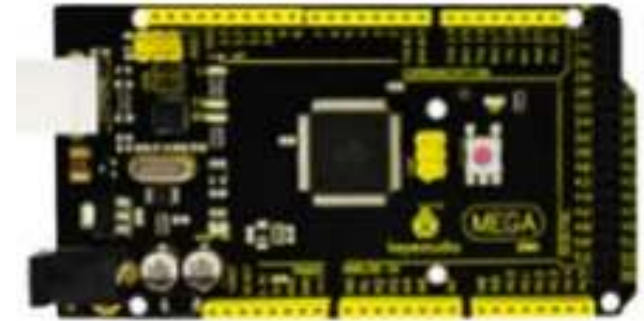
На схеме слева — 12, 9, 8 и 6 — пины-аноды для каждого разряда, на схеме справа — соответствие пинов индикатора его сегментам.

# Проект на Arduino

Мы можем приступить к нашему проекту.

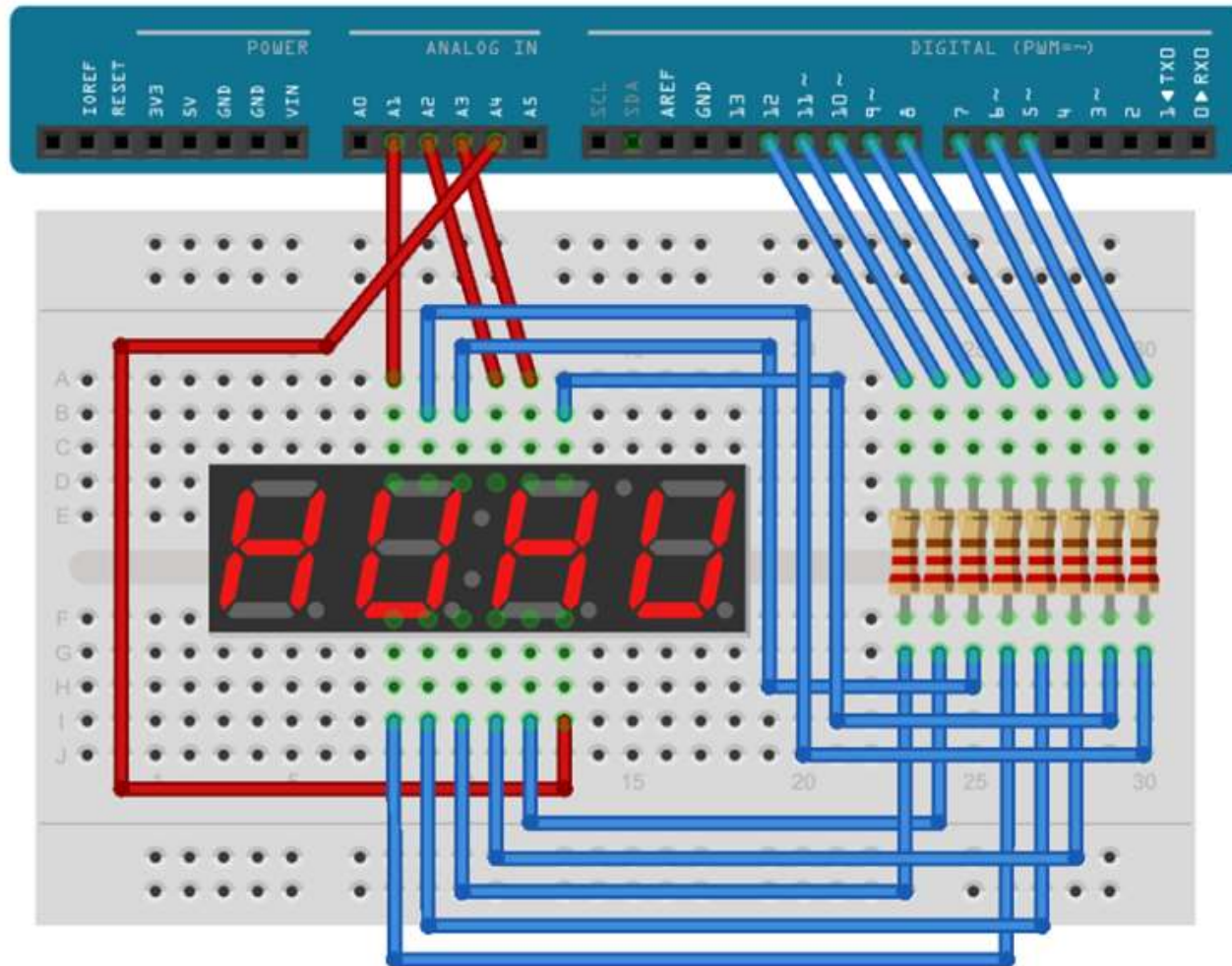
Для этого нам понадобится:

- Плата Arduino Mega 2560;
- Макетная плата (брэдборд);
- 20 проводов папа-папа;
- 8 резисторов 220 Ом;
- 4х рядный 7-ми сегментный индикатор.



# Проект на Arduino

Наш проект должен выглядеть следующим образом:



# Начало скетча

```
int anodPins[] = {A1, A2, A3, A4}; // Задаем пины для каждого разряда
int segmentsPins[] = {5, 6, 7, 8, 9, 10, 11, 12}; //Задаем пины для каждого сегмента (из 7 + 1(точка))
```

```
void setup() {
  // Все выходы программируем как OUTPUT
  for (int i = 0; i < 4; i++) {
    pinMode(anodPins[i], OUTPUT);
  }
  for (int i = 0; i < 8; i++) {
    pinMode(segmentsPins[i], OUTPUT);
  }
}
```

//{A, B, C, D, E, F, G,DP} - распиновка сегментов

```
int seg[10][8] = {
  {1, 1, 1, 1, 1, 1, 0, 0}, //Цифра 0
  {0, 1, 1, 0, 0, 0, 0, 0}, //Цифра 1
  {1, 1, 0, 1, 1, 0, 1, 0}, //Цифра 2
  {1, 1, 1, 1, 0, 0, 1, 0}, //Цифра 3
  {0, 1, 1, 0, 0, 1, 1, 0}, //Цифра 4
  {1, 0, 1, 1, 0, 1, 1, 0}, //Цифра 5
  {1, 0, 1, 1, 1, 1, 1, 0}, //Цифра 6
  {1, 1, 1, 0, 0, 0, 0, 0}, //Цифра 7
  {1, 1, 1, 1, 1, 1, 1, 0}, //Цифра 8
  {1, 1, 1, 1, 0, 1, 1, 0} //Цифра 9
};
```

Продолжение дальше

# Продолжение скетча

```
int t = 0;
int digid = 0;
void loop() {
  t += 1;
  if (t > 9999) t = 0;
  if ((t % 1000) == 0) {
    digid = t / 1000; //Каждую секунду отображаем цифры подряд
  }
  for (int i = 0; i < 4; i++) { // Каждый разряд по очереди
    for (int k = 0; k < 8; k++) { // Каждый сегмент по очереди - исходя из заданной
карты
      digitalWrite(segmentsPins[k], ((seg[digid][k] == 1) ? LOW : HIGH));
    }
    digitalWrite(anodPins[i], HIGH);
    delay(1);
    digitalWrite(anodPins[i], LOW);
  }
}
```

# Контрольные вопросы

- Как подключается 7-ми сегментный индикатор?
- Перечислите выводы индикатора?
- Что можно выводить на данном индикаторе?