

Практическая работа 13

Секундомер

В этом эксперименте мы создаем секундомер, который считает до 10.

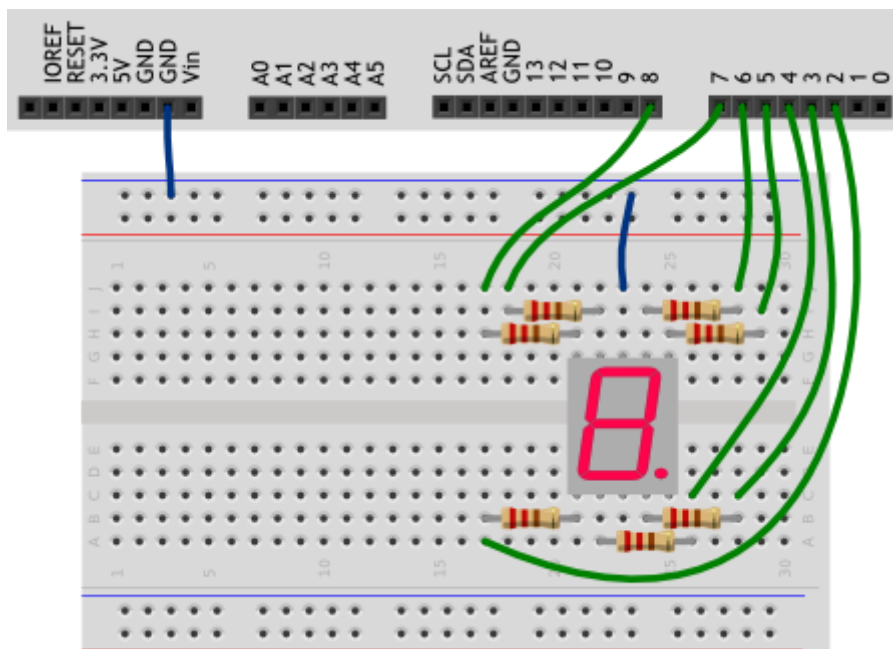
Задание 1. Ответить на вопросы

1. Что такое семисегментный индикатор?
2. Где применяется семисегментный индикатор?

Задание 2. Список деталей для эксперимента

1. 1 плата Arduino Uno
2. 1 беспаячная макетная плата
3. 1 семисегментный индикатор
4. 7 резисторов 220 Ом
5. 9 проводов «папа-папа»

Схема на макетной плате:



1. Зарисуйте принципиальную схему установки.

Обратите внимание

✓ Выводы 3 и 8 семисегментного индикатора оба являются катодами, к земле можете подключать любой из них.

✓ Внимательно рассмотрите схему, сопоставьте сегменты индикатора с номерами его ножек, а те, в свою очередь, с пинами Arduino, к которым мы их подключаем.

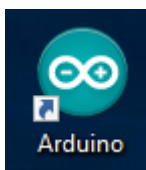
✓ Вывод 5 индикатора — это точка. Мы не используем её в этом эксперименте

✓ Сегменты индикатора — просто светодиоды, поэтому мы используем резистор с каждым из них.

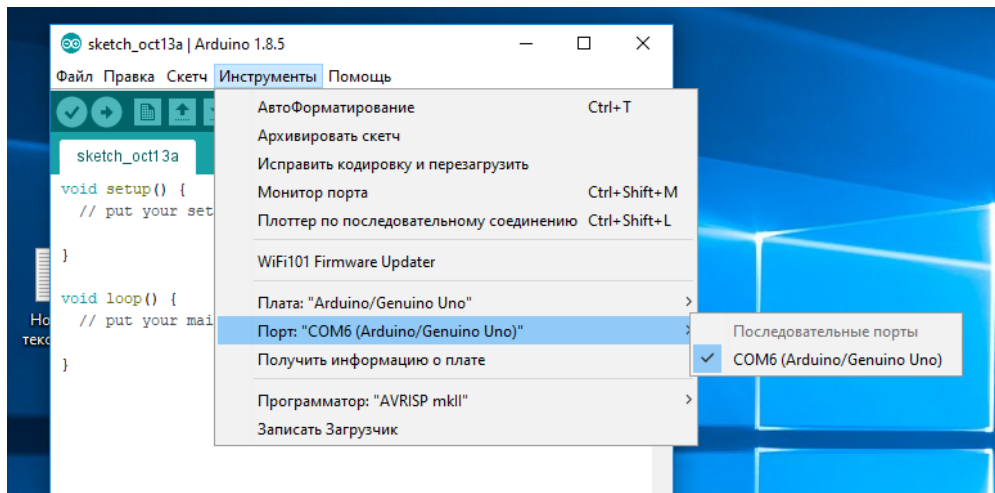
✓ Сегменты китайского индикатора могут отличаться от представленного

Задание 3. Программирование микроконтроллера

1. Запустите приложение



2. Убедитесь, что выбран нужный порт



3. Наберите в редакторе кода следующий код программы:

```

sketch_nov20a $
#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] = {
  0b00111111, 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);
  }
}

```

Пояснения к коду

✓ Мы создали массив типа byte: каждый его элемент это 1 байт, 8 бит, может принимать значения от 0 до 255.

✓ Символы арабских цифр закодированы состоянием пинов, которые соединены с выводами соответствующих сегментов: 0, если сегмент должен быть выключен, и 1, если включен.

✓ В переменную mask мы помещаем тот элемент массива numberSegments, который соответствует текущей секунде, вычисленной в предыдущей инструкции.

- ✓ В цикле for мы пробегаем по всем сегментам, извлекая с помощью встроенной функции bitRead нужное состояние для текущего пина, в которое его и приводим с помощью digitalWrite и переменной enableSegment
- ✓ bitRead(x, n) возвращает boolean значение: n-ый бит справа в байте x

Задание 4. Ответьте на следующие вопросы

1. К какой ножке нашего семисегментного индикатора нужно подключать землю?
2. Как мы храним закодированные символы цифр?
3. Каким образом мы выводим символ на индикатор?

Задание 5. Самостоятельно измените существующую программу и схему

1. Измените код, чтобы индикатор отсчитывал десятые секунды.
2. Поменяйте программу так, чтобы вместо символа «0» отображался символ «A».
3. Дополните схему и программу таким образом, чтобы сегмент-точка включался при прохождении четных чисел и выключался на нечетных