Практическая работа 18

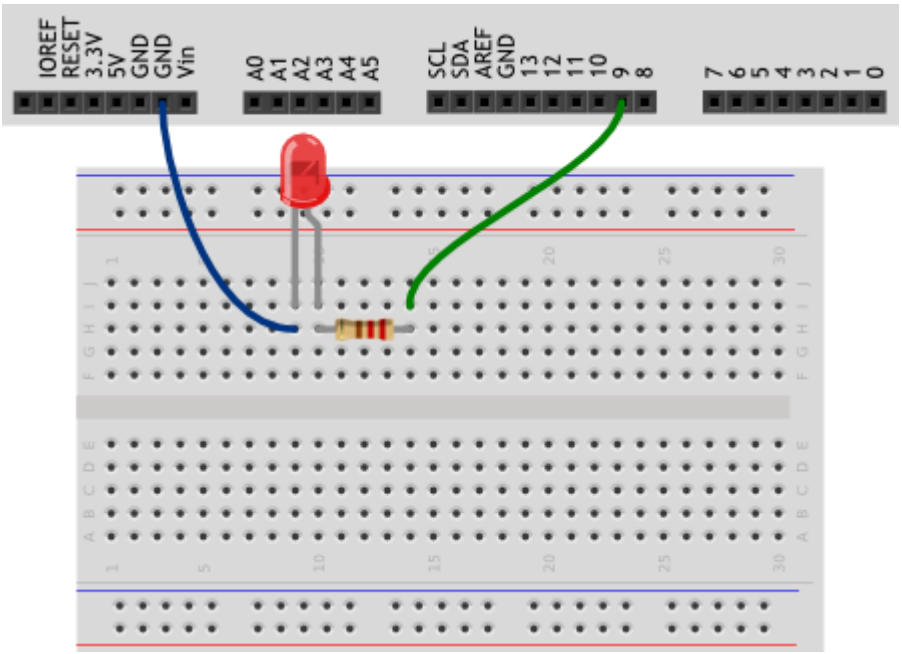
*Светильник, управляемый по USB*

В этом эксперименте мы отправляем устройству команды, как ему светить.

## Задание 2. Список деталей для эксперимента

1. 1 плата Arduino Uno
2. 1 беспаечная макетная плата
3. 1 резистор 220 Ом
4. 2 проводов «папа-папа»

Схема на макетной плате:

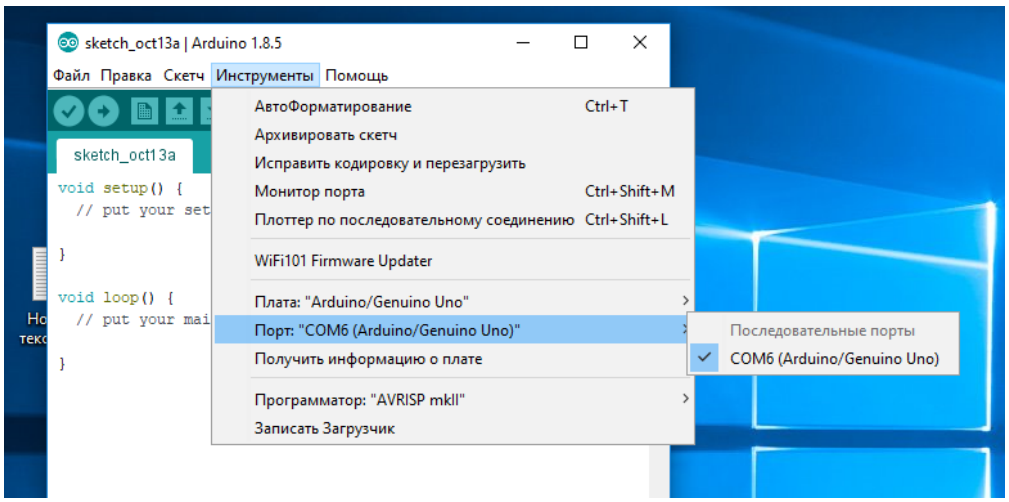



1. Зарисуйте принципиальную схему установки.

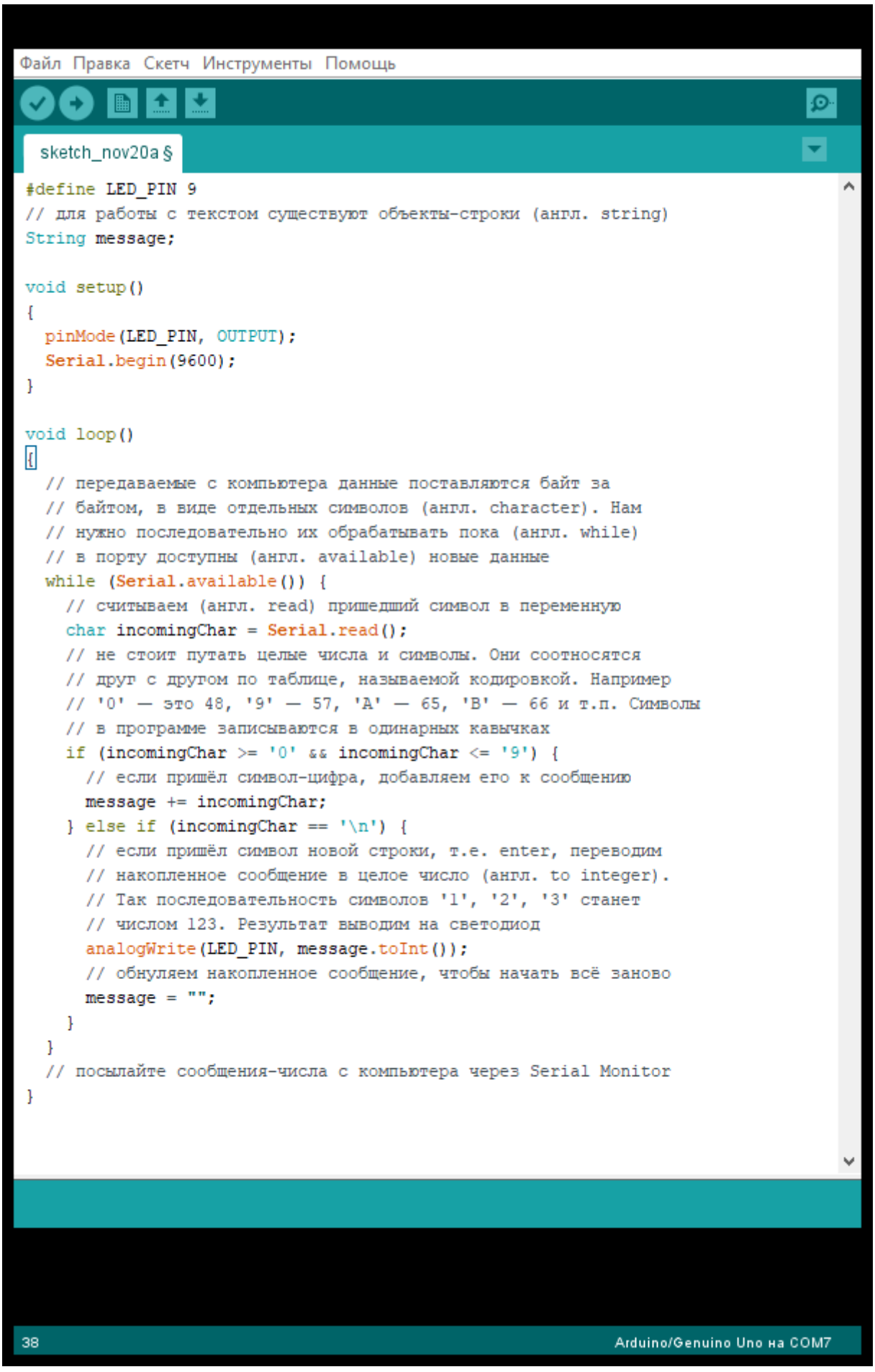
## Задание 3. Программирование микроконтроллера

1. Запустите приложение


2. Убедитесь, что выбран нужный порт




3. Наберите в редакторе кода следующий код программы:


Файл Правка Скетч Инструменты Помощь
sketch_nov20a §
#define LED_PIN 9
// для работы с текстом существуют объекты-строки (англ. string)
String message;
void setup()
{
pinMode(LED_PIN, OUTPUT);
Serial.begin(9600);
}
void loop()
{
// передаваемые с компьютера данные поставляются байт за
// байтом, в виде отдельных символов (англ. character). Нам
// нужно последовательно их обрабатывать пока (англ. while)
// в порту доступны (англ. available) новые данные
while (Serial.available()) {
// считываем (англ. read) пришедший символ в переменную
char incomingChar = Serial.read();
// не стоит путать целые числа и символы. Они соотносятся
// друг с другом по таблице, называемой кодировкой. Например
// '0' — это 48, '9' — 57, 'A' — 65, 'B' — 66 и т.п. Символы
// в программе записываются в одинарных кавычках
if (incomingChar >= '0' && incomingChar <= '9') {
// если пришёл символ-цифра, добавляем его к сообщению
message += incomingChar;
} else if (incomingChar == '\n') {
// если пришёл символ новой строки, т.е. enter, переводим
// накопленное сообщение в целое число (англ. to integer).
// Так последовательность символов '1', '2', '3' станет
// числом 123. Результат выводим на светодиод
analogWrite(LED_PIN, message.toInt());
// обнуляем накопленное сообщение, чтобы начать всё заново
message = "";
}
}
// посылайте сообщения-числа с компьютера через Serial Monitor
}
38
Arduino/Genuino Uno на COM7

**Пояснения к коду**

✓ В этой программе мы создаем объект класса String. Это встроенный класс, предназначенный для работы со строками, т.е. с текстом.

✓ Не путайте его с типом данных string, который является просто массивом символов. String же позволяет использовать ряд методов для удобной работы со строками.

✓ Мы знакомимся с новым видом циклов: цикл с условием while. В отличие от цикла со счетчиком for, цикл while(expression) выполняется до тех пор, пока логическое выражение expression истинно.

✓ Метод available() объекта Serial возвращает количество байт, полученных через последовательный порт.

✓ В данном эксперименте цикл while работает до тех пор, пока available() возвращает ненулевое значение, любое из которых приводится к true.

✓ Переменные типа char могут хранить один символ.

✓ В этом примере символ мы получаем методом Serial.read(), который возвращает первый байт, пришедший на последовательный порт, или -1, если ничего не пришло.

✓ Обратите внимание, что в if мы сравниваем не пришедший символ с 0 и 9, но их коды. Если пришел какой-то символ, который не является цифрой, мы не будем его добавлять к нашей строке message.

✓ Объекты типа String позволяют производить конкатенацию, т.е. объединение строк. Это можно сделать так: message = message + incomingChar, но можно записать в сокращенной форме: message += incomingChar.

✓ В этой программе мы дополняем if конструкцией else if. Это еще один условный оператор, который проверяется только в случае ложности выражения, данного первому оператору. Несколько else ifмогут следовать друг за другом, при этом каждое следующее условие будет проверяться только в случае невыполнения всех предыдущих. Если в конце разместить else, он выполнится только если ни одно из условий не выполнено.

✓ Напомним, что последовательностью \n кодируется символ переноса строки. Если он был передан устройству, мы передаем полученные ранее символы как параметр для analogWrite(), которая включает светодиод.

✓ Мы используем один из методов String, toInt(), который заставляет считать строку не набором цифр, но числом. Он возвращает значение типа long, при этом, если строка начинается с символа, не являющегося цифрой, будет возвращен 0. Если после цифр, идущих в начале строки, будут символы не-цифры, на них конверсия остновится.

✓ Обратите внимание на выпадающее меню внизу монитора порта: чтобы наше устройство получало символ перевода строки, там должно быть выбрано «Новая строка (NL)»

✓ Пустая строка обозначается так: "". Опустошив ее, мы готовы собирать новую последовательность символов.

Задание 4. Ответьте на следующие вопросы

1. Какие объекты позволяют легко манипулировать текстовыми данными?
2. Что возвращают методы Serial.available() и Serial.read()?
3. Чем отличаются конструкции for и while?
4. Каким образом можно организовать более сложное ветвление, чем if … else?
5. Как можно объединить текстовые строки?
6. Как можно привести текстовую строку, содержащую цифры, к числовому типу?

Задание 5. Самостоятельно измените существующую программу и схему

1. Проверьте, попадает ли переданное число в диапазон значений, которые нужно передавать в analogWrite(). Передайте на компьютер сообщение об ошибке, если нет.

2. Переделайте программу так, чтобы устройство распознавало текстовые команды, например, «on» и «off», и соответственно включало и выключало светодиод.

Вам может пригодиться один из методов String: toLowerCase(yourString) или toUpperCase(yourString), которые возвращают переданную строку yourString, приведенную к нижнему или верхнему регистру соответственно.