

### Лабораторная работа 3. Отображение данных из Arduino в окне монитора последовательного порта

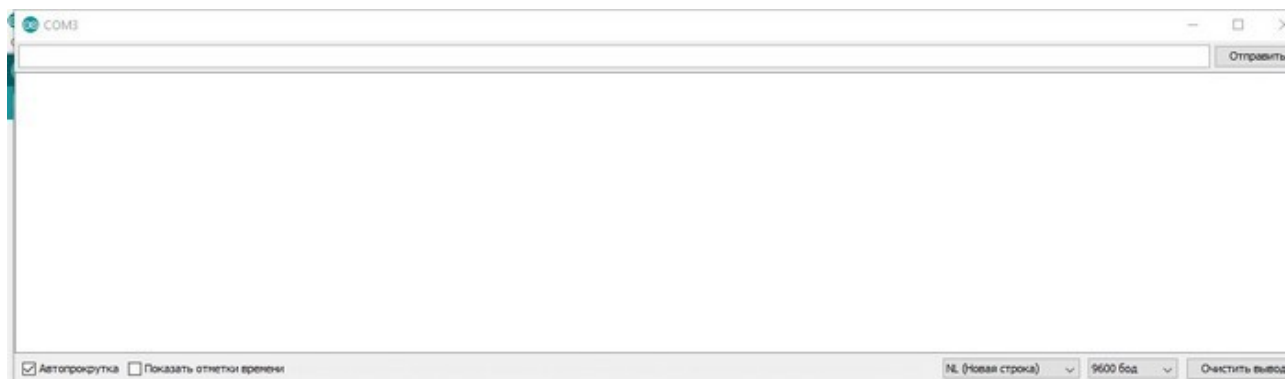
На этом занятии вы узнаете, как использовать проводное соединение Arduino с компьютером и окно монитора порта в IDE для вывода данных с платы и отправки данных в плату с клавиатуры.

#### Монитор последовательного порта

Чтобы открыть монитор порта, запустите IDE и щелкните на пиктограмме "Монитор порта" на панели инструментов.



В ответ на это откроется окно монитора порта.



Вверху в окне монитора порта имеется однострочное поле ввода с кнопкой Отправить и область вывода под ним, где отображаются данные, полученные с платы Arduino. Если поставить флажок Автопрокрутка, в области вывода будут отображаться самые свежие данные, а при ее переполнении старые данные будут «уходить» за границы области. Сняв флажок Автопрокрутка, вы сможете вручную прокручивать область вывода во время исследования данных.

#### Инициализация обмена с монитором порта

Прежде чем использовать монитор порта, его нужно активировать, добавив следующий вызов в функцию `void setup()`:

```
Serial.begin(9600);
```

Число 9600 — это скорость передачи данных между компьютером и платой Arduino, измеряемая в бодах. Это число должно соответствовать

значению, выбранному в раскрывающемся списке справа внизу в окне монитора порта.

### **Отправка текста в монитор порта**

Послать текст для отображения в области вывода в мониторе порта можно вызовом функции `Serial.print`:

```
Serial.print("Arduino!");
```

Она отправит в монитор порта текст, заключенный в кавычки.

Для вывода текста и принудительного перехода на новую строку воспользуйтесь функцией `Serial.println`:

```
Serial.println("Arduino!");
```

### **Вывод содержимого переменных**

Аналогично осуществляется вывод в монитор порта содержимого переменных. Например, далее показано, как вывести содержимое переменной `results`: `Serial.println(results)`;

Значения переменных типа `float` по умолчанию выводятся с двумя знаками после десятичной точки. Количество знаков можно изменить, передав число от 0 до 6 во втором параметре после имени переменной. Например, чтобы вывести значение вещественной переменной `results` с четырьмя десятичными знаками, выполните следующий вызов: `Serial.print(results,4)`;

### **Задание 1. Передача данных из Arduino в «Монитор порта»**

Для того чтобы передать данные из Arduino в «Монитор порта» необходимо в функции `setup()` его нужно активировать, добавив следующий вызов в функцию `void setup()`: `Serial.begin(9600)`.

```
1 void setup()
2 {
3   Serial.begin(9600);
4
5 }
6
7 void loop()
8 {
9   Serial.println("Arduino");
10  delay(1000); // Wait for 1000 millisecond(s)
11 }
```

Таким образом, в мониторе порта мы будем видеть строку "Arduino" каждую секунду .

## Монитор последовательного интерфейса

```
Arduino  
Arduino  
Arduino  
Arduino
```

Класс Serial содержит и другие методы для работы с монитором порта, но, как правило, методов print и println достаточно в большинстве случаев.

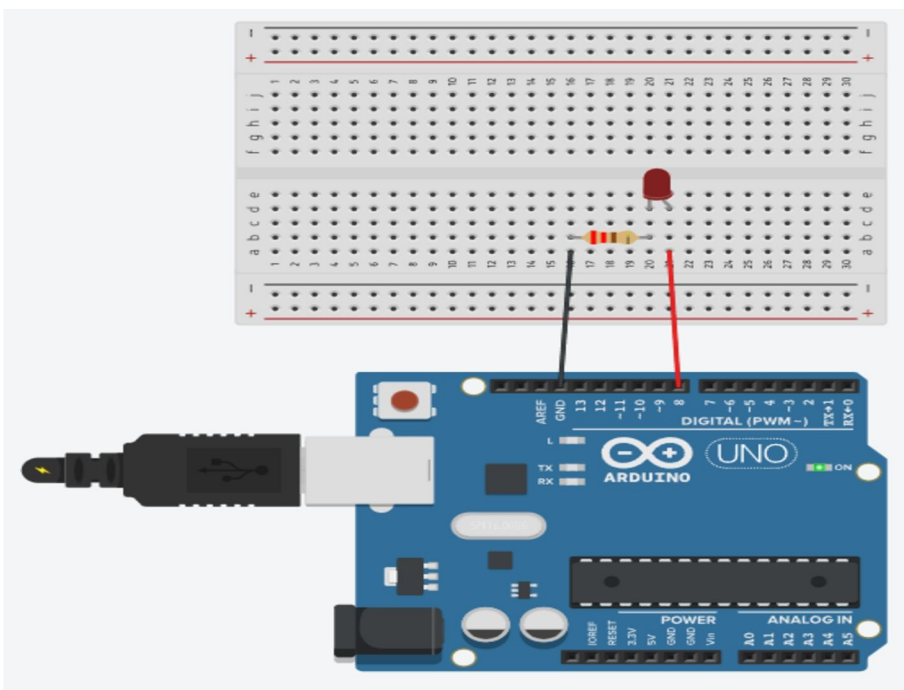
### Задание 2. Получение данных, отправленных из «Монитора порта» в Arduino

Теперь мы попробуем получить команды, отправленные из монитора порта в Arduino и использовать их, например, для включения или выключения светодиода.

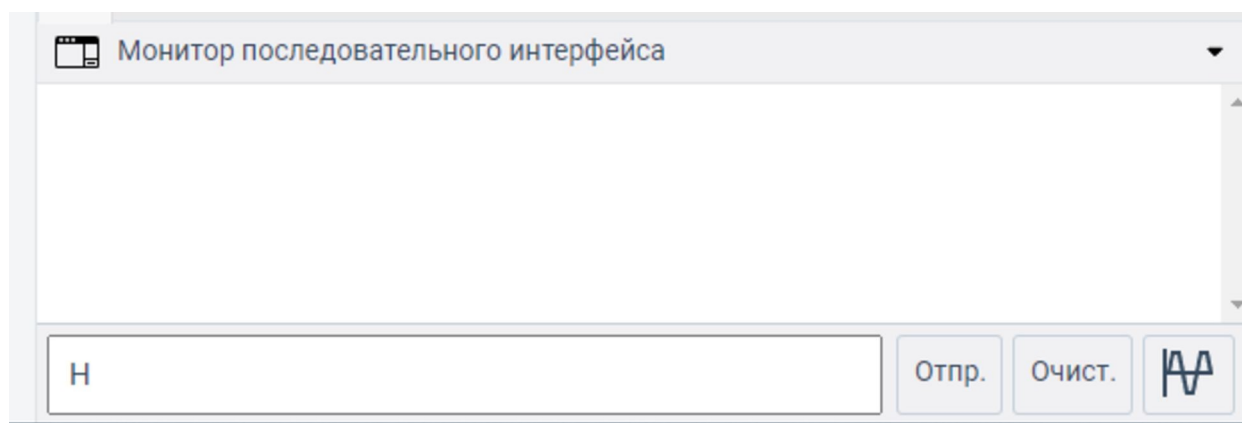
И так, по команде "H" мы будем включать светодиод, а по команде "L" будем выключать.

Для того, чтобы отправить команду, необходимо в поле расположенном в верхней части формы набрать текст, и нажать кнопку «Отправить».

Должно получиться следующее.



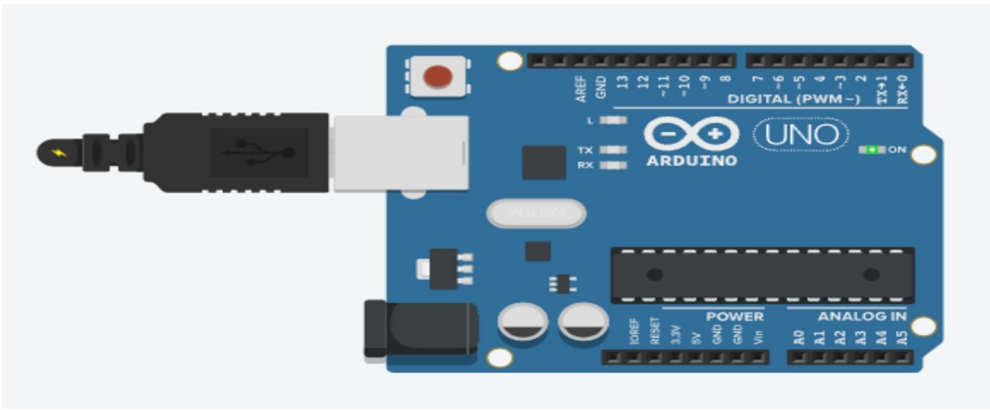
```
1 char ch;
2
3 void setup()
4 {
5   Serial.begin(9600);
6   pinMode(8,OUTPUT);
7
8 }
9
10 void loop()
11 {
12   if (Serial.available() > 0) {
13     ch = Serial.read();
14     if (ch == 'H') {
15       digitalWrite(8,HIGH);
16     }
17     if (ch == 'L') {
18       digitalWrite(8,LOW);
19     }
20     delay(1000);
21   }
22 }
23
```



**Задание.** Собрать схему на монтажной плате, перенести программу в систему программирования Arduino IDE и провести эксперимент.

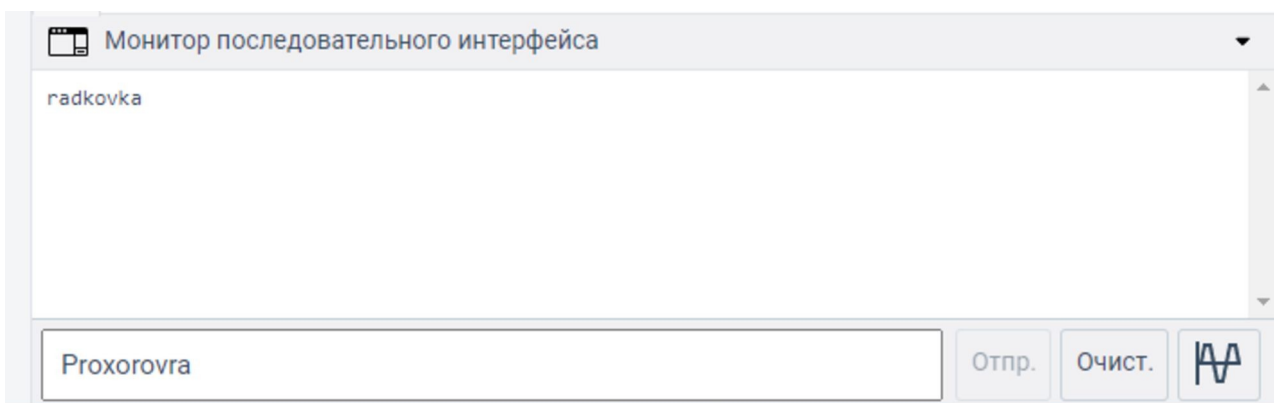
**Задание 3.** Печать данных, отправленных из «Монитора порта» в «Монитора порта»

Данные вводятся в компьютер из«Монитора порта» и выводятся на «Монитор порта»



Изменить описание переменной `val` на символьный тип `char`.

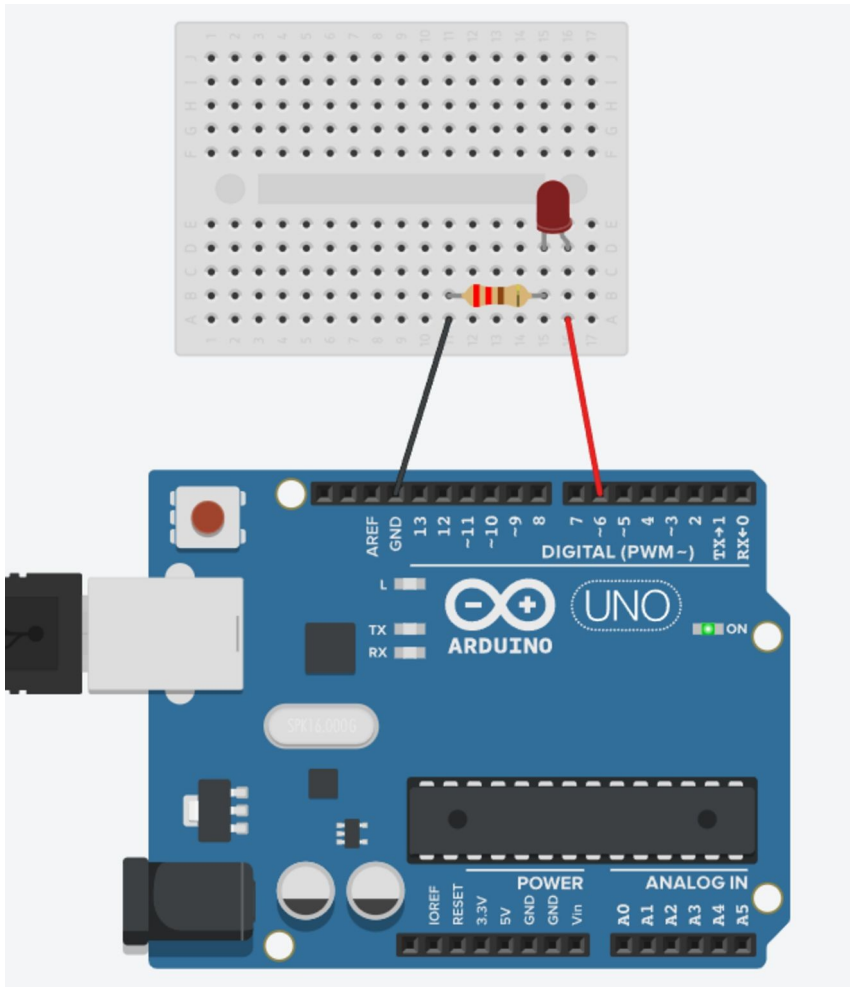
```
Текст 1 (Arduino Uno R3)
1 char val;
2
3 void setup()
4 {
5   Serial.begin(9600); // Установка порта на скорость 9600 бит/сек
6
7 }
8
9 void loop()
10 {
11   if (Serial.available() > 0) // Если есть символ во входном потоке,
12   {
13     val = Serial.read(); // то читаем его и сохраняем в val
14     Serial.print(val); // печатаем в порт
15   }
16   delay(10); // задержка 10 мс
17 }
```



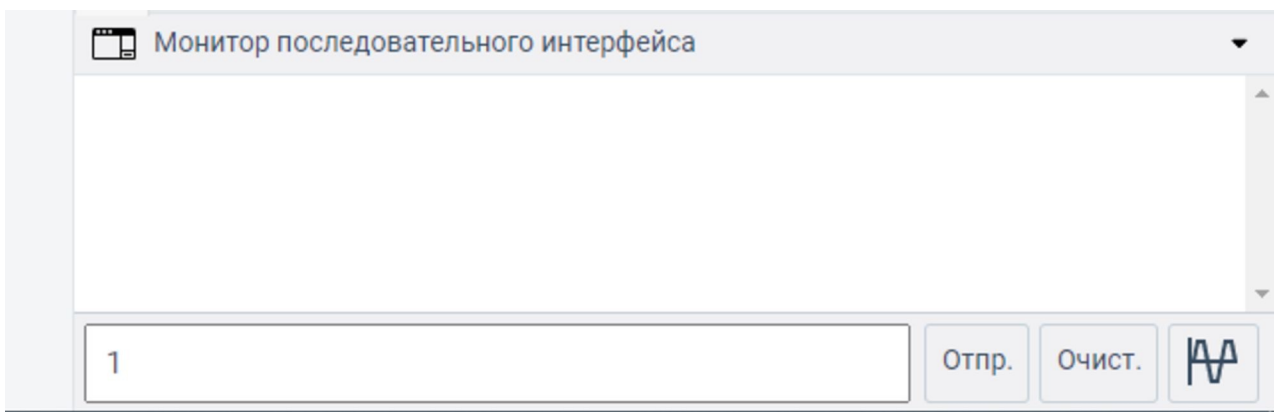
**Задание.** Собрать схему на монтажной плате, перенести программу в систему программирования Arduino IDE и провести эксперимент.

**Задание 4.** Условный оператор, управление миганием светодиодом с ПК

Создать схему с одним светодиодом. Если с "Монитора порта" вводится символ "1", то лампочка горит 3 сек., "2" - горит 1 сек., "3" - горит 0,1 сек. Начальное время задается равным 50 мс.



```
1 int time = 50;
2
3 char val;
4
5 void setup()
6 {
7   Serial.begin(9600);
8
9   pinMode(6, OUTPUT);
10 }
11
12 void loop()
13 {
14   if (Serial.available() > 0) {
15     val = Serial.read();
16     if (val == '1') {
17       time = 3000;
18     }
19     if (val == '2') {
20       time = 1000;
21     }
22     if (val == '3') {
23       time = 100;
24     }
25
26     digitalWrite(6, HIGH);
27     delay(time); // Wait for time millisecond(s)
28     digitalWrite(6, LOW);
29     delay(time); // Wait for time millisecond(s)
30   }
31 }
```



**Задание.** Собрать схему на монтажной плате, перенести программу в систему программирования Arduino IDE и провести эксперимент.