***То, что сегодня наука, – завтра техника***

**Эдвард Теллер**

Практические работы по робототехнике выполненные с использованием Arduino

Сборник

****Мыски 2024

Автор - составитель: учитель технологии Копильчак Евгения Олеговна

Практические работы по робототехнике выполненные с использованием Arduino

Сборник. – Мыски: МБОУ «СОШ№5»,

2024. –39с.

В сборник включены практические работы и принципиальные схемы для Arduino.

****Сборник адресован тем ребятам, которые заинтересованы в робототехнике, но не знают, как и с чего начать.

**Содержание**

Введение…………………….……………………4

Практическая 1. Светодиод…………………..…5

Практическая 2. Упрвляемый «программно» светодиод………………………………………….…8

Практическая 3. Светодиод, управляемый «вручную»……………………………………..............11

Практическая 4. Пьезодинамик……………..…14

Практическая 5. Тактовая кнопка……………...17

Практическая 6. Семисегментный индикатор………………………………………………..20

Практическая 7. LCD дисплей…………………23

Практическая 8. Сервопривод………………….28

Практическая 9. Модуль звука с микрофоном..32

Практическая 10. Светофор…………….………35

Заключение……………………………….……..39

****

**Введение**

На сегодняшний день робототехника является очень актуальной дисциплиной, так как роботы разных видов используются во всех отраслях промышленности, в авиации, строительстве и быту. Наиболее сложные устройства необходимы для экстремальных сфер деятельности: военной, подводной и космической.

Для того что бы учащиеся могли реализовать весь свой творческий потенциал и свободно ориентироваться в современных технологиях необходимо изучать робототехнику на внеурочной деятельности.

Данный сборник будет полезен для тех ребят, которые не разу не сталкивались с робототехникой, поможет изучить основные практические работы на самых простых и часто используемых элементах электронного конструктора Arduino.

**Практическая 1.Светодиод**

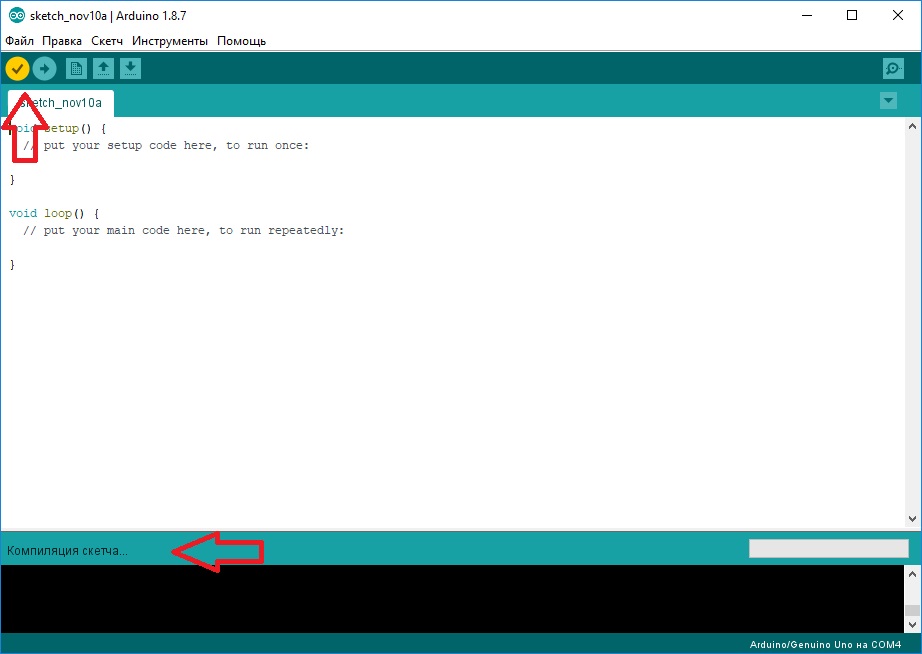
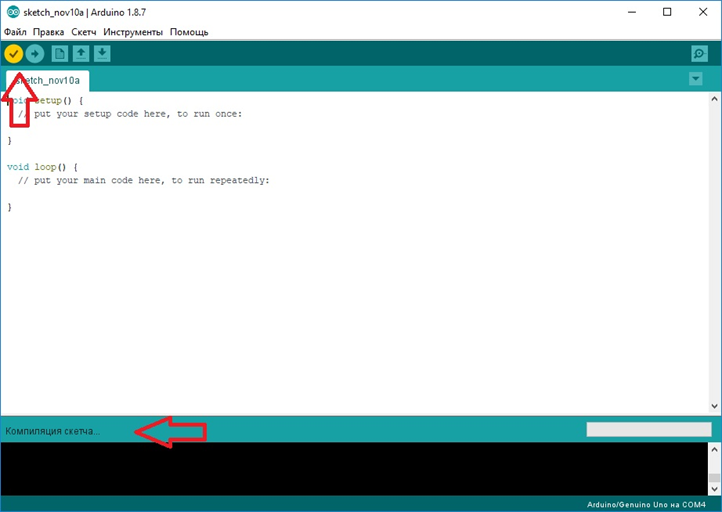
***Цель работы:***

Создания программы мигания светодиода с заданной периодичностью с помощью резистора и светодиода.

**Оборудование**

* Микроконтроллер- ArduinoUNO
* Макетная плата
* Резистор
* Светодиод
* Соединительные провода «папа-папа»
* USB кабель
* компьютер

**Ход работы**

1. Собрать схемуРисунок 1 – Принципиальная схема подключения светодиода.
2. На компьютере запустить программу Arduinо;
3. Прописать программу
4. Проверить программу
5. Подключить через USB кабель
6. Загрузить программу на микрокроконтроллер

***Схема подключения:***

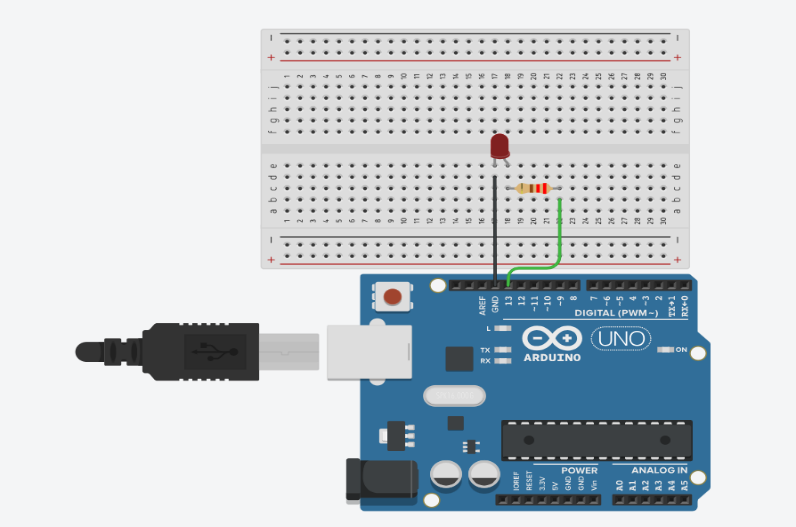


Рисунок 1 – Принципичальная схема подключения светодиода

***Скетч***

// C++ code

//

/\*

This program blinks pin 13 of the Arduino (the

built-in LED)

\*/

void setup()

{

pinMode(LED\_BUILTIN, OUTPUT);

}

void loop()

{

// turn the LED on (HIGH is the voltage level)

digitalWrite(LED\_BUILTIN, HIGH);

delay(1000); // Wait for 1000 millisecond(s)

// turn the LED off by making the voltage LOW

digitalWrite(LED\_BUILTIN, LOW);

delay(1000); // Wait for 1000 millisecond(s)



**Практическая 2.Управляемый «программно» светодиод**

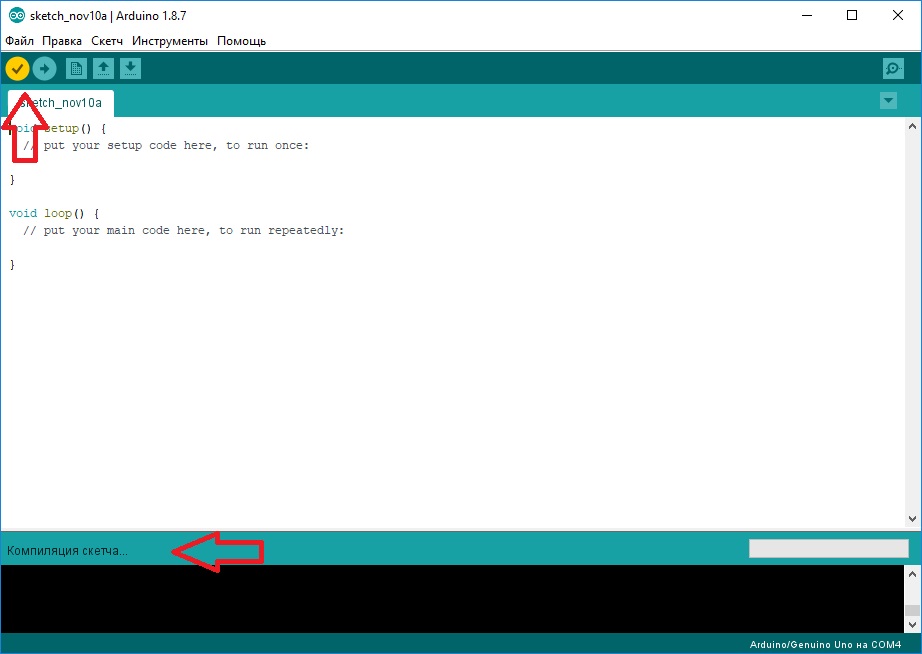
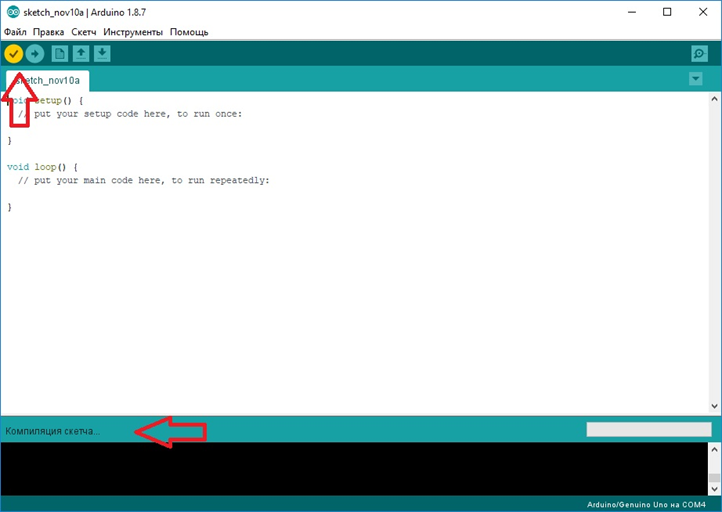
***Цель работы:***

Создание программы управления яркости светодиода с заданной «программно», периодичностью.

**Оборудование**

* Микроконтроллер- Arduino UNO
* Макетная плата
* Резистор
* Светодиод
* Соединительные провода «папа-папа»
* USB кабель
* Компьютер

**Ход работы**

1. Собрать схему рисунок 2;
2. На компьютере запустить программу Arduinо;
3. Прописать программу
4. Проверить программу
5. Подключить через USB кабель
6. Загрузить программу на микрокроконтроллер

***Схема подключения:***

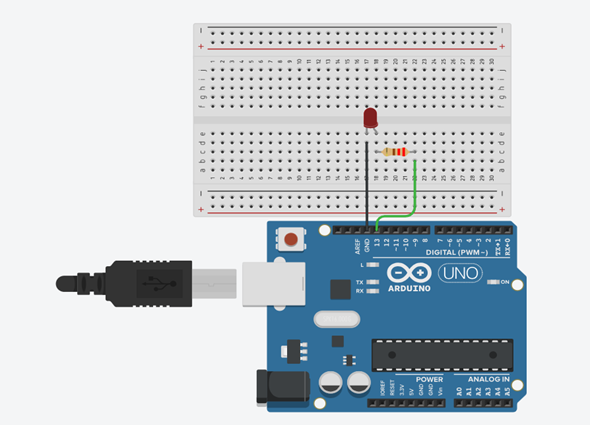
******

Рисунок 2 – Принципиальная схема подключения светодиода

***Скетч***

int led = 9;

void setup()

{

pinMode(led, OUTPUT);

}

void loop()

{

//3 Вольта

analogWrite(led, 153);

delay(1000);

//1,5 Вольта

analogWrite(led, 76);

delay(1000);

//5 Вольт

analogWrite(led, 255);

delay(1000);

}



**Практическая 3.Светодиод, управляемый «вручную»**

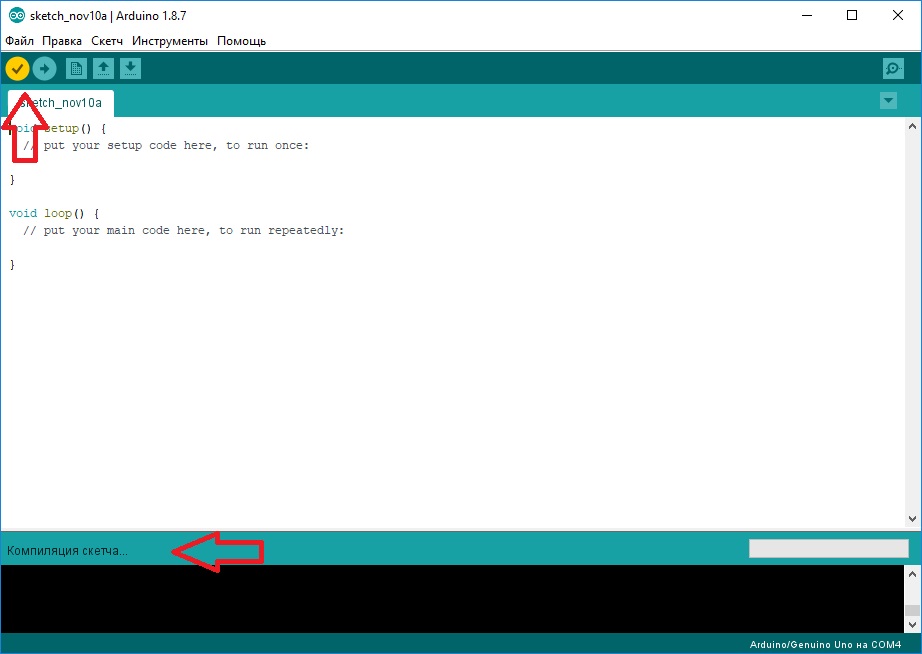
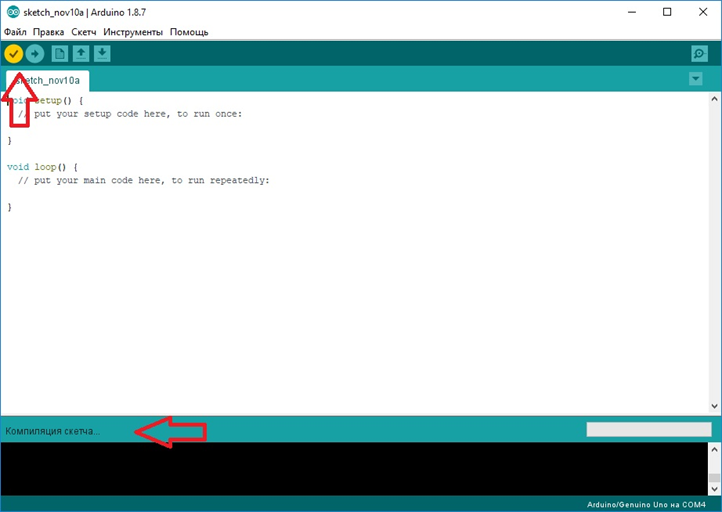
***Цель работы:***

Создание программы управления яркости светодиода «вручную», используя значение напряжения, выставляемого потенциометра.

**Оборудование**

* Микроконтроллер- Arduino UNO
* Макетная плата
* Резистор
* Светодиод
* Соединительные провода «папа-папа»
* USB кабель
* Компьютер
* Потенциометр

**Ход работы**

1. Собрать схему рисунок 3;
2. На компьютере запустить программу Arduinо;
3. Прописать программу
4. Проверить программу
5. Подключить через USB кабель
6. Загрузить программу на микрокроконтроллер
7. покрутите ручкой потенциометра – яркость светодиода должна изменятся

***Схема подключения:***

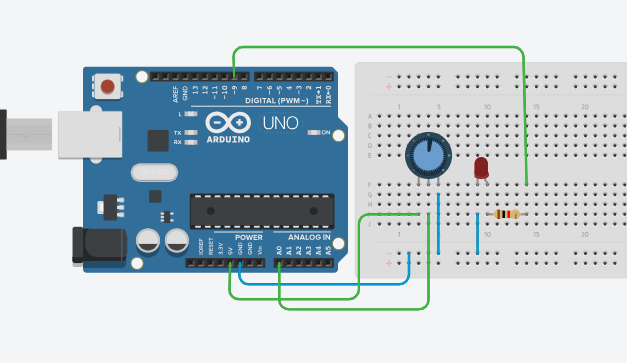
******

Рисунок 3 – Принципиальная схема подключения светодиода и потенциометра

***Скетч***

int led = 9;

int pot = A0;

void setup()

{

pinMode(led, OUTPUT);

pinMode(pot, INPUT);

}

void loop()

{

int x;

x = analogRead(pot)/4;

analogWrite(led, x);

}

****

**Практическая 4.Пьезодинамик**

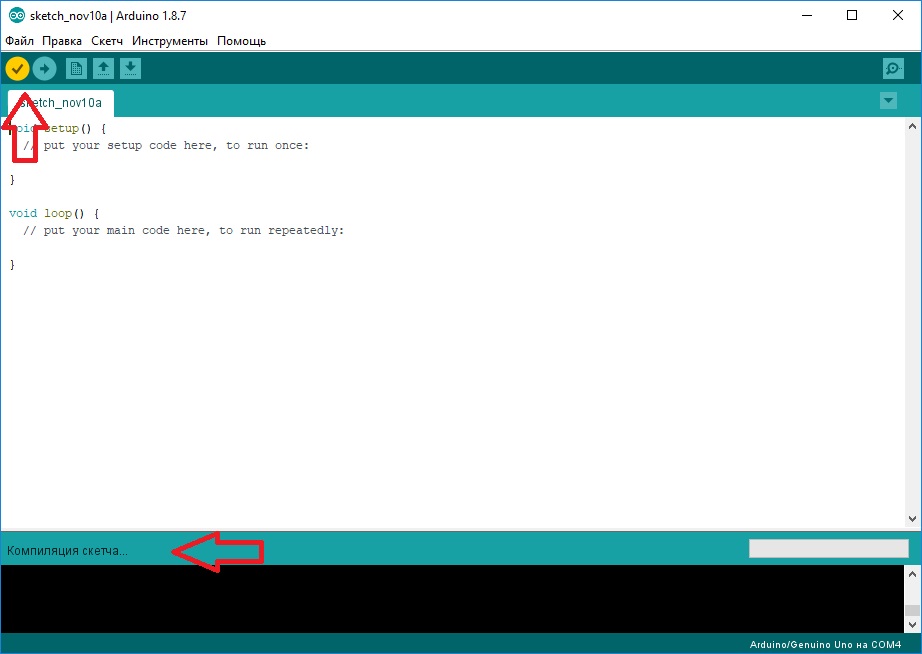
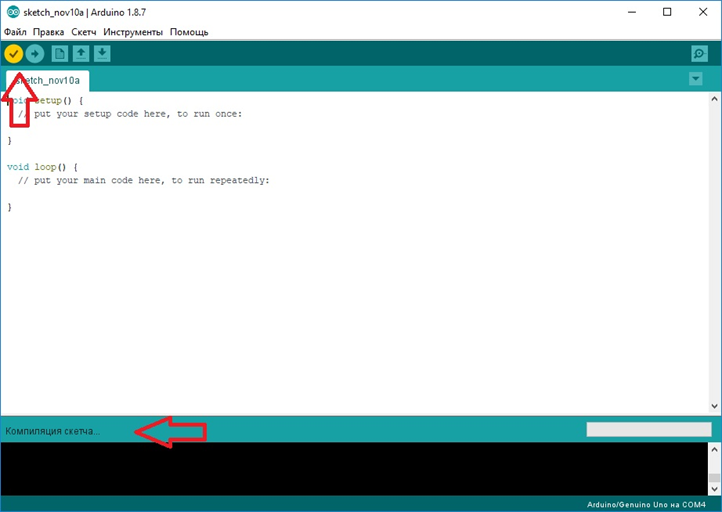
***Цель работы:***

Создание программы по управлениюзвучания пьезодинамика.

**Оборудование**

* Микроконтроллер- Arduino UNO
* Макетная плата
* Соединительные провода «папа-папа»
* USB кабель
* Компьютер
* Пьезодинамик

**Ход работы**

1. Собрать схему рисунок 3
2. На компьютере запустить программу Arduinо;
3. Прописать программу
4. Проверить программу
5. Подключить через USB кабель
6. Загрузить программу на микрокроконтроллер

***Схема подключения:***

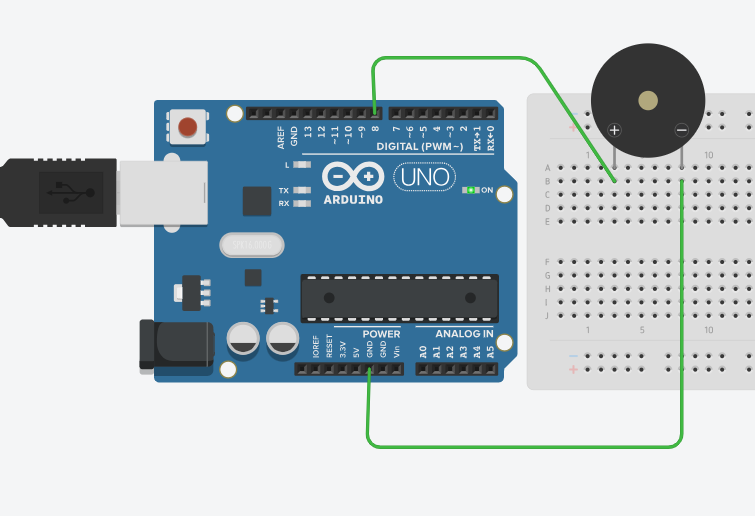
******

Рисунок 1 – Принципиальная схема подключения светодиода

***Скетч***

void setup()

{

pinMode(3, OUTPUT);

pinMode(A10,INPUT);

Serial.begin(9600);

}

void loop()

{

int x = analogRead(A10);

inty = map(x,0, 1023, 3500, 4500);// преобразованиесигнала

Serial.println(y); // печатается значение частоты сигнала

tone(3, y, 20); // задается длительность сигнала 20 мс

}



**Практическая 5.Тактовая кнопка**

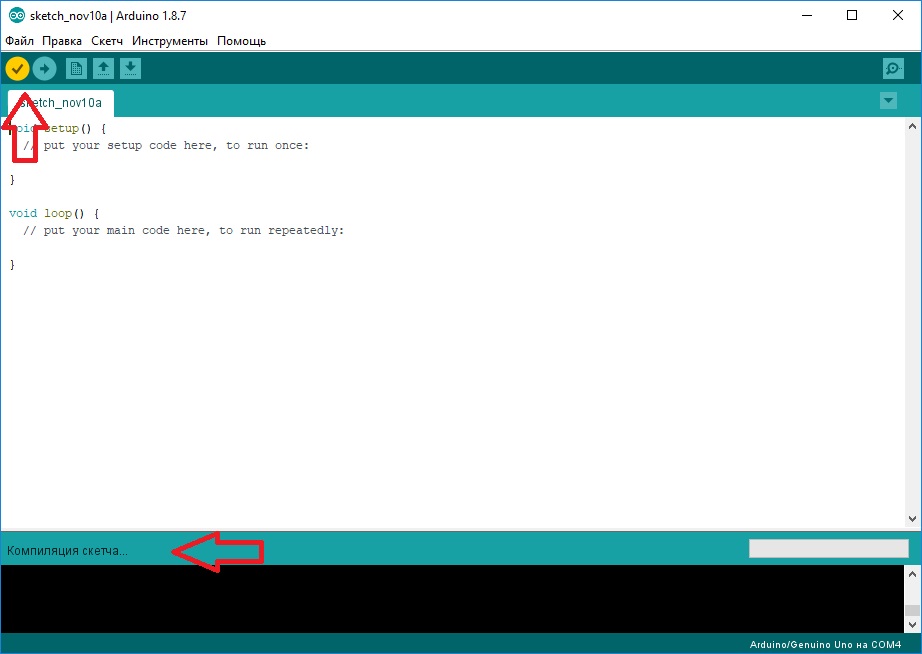
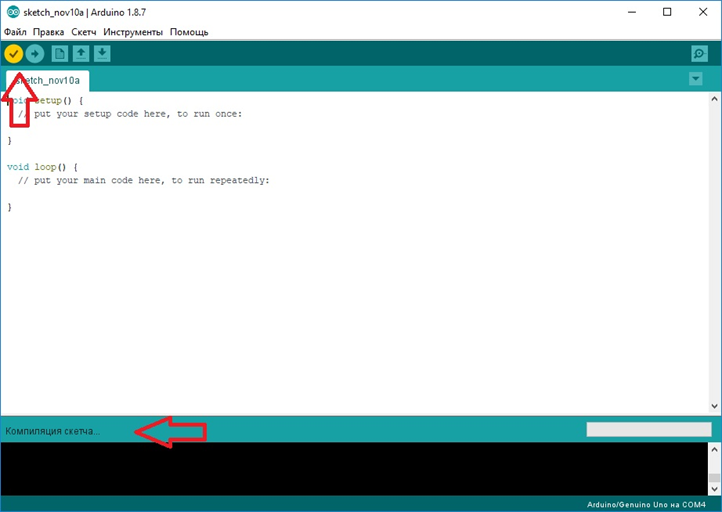
***Цель работы:***

Создание программы по управлению включением и выключением светодиода с помощью кнопки.

**Оборудование**

* Микроконтроллер- Arduino UNO
* Макетная плата
* Резистор (2 шт)
* Светодиод
* Соединительные провода «папа-папа»
* USB кабель
* Компьютер
* Тактовая кнопка

**Ход работы**

1. Собрать схему рисунок 4;
2. На компьютере запустить программу Arduinо;
3. Прописать программу
4. Проверить программу
5. Подключить через USB кабель
6. Загрузить программу на микрокроконтроллер
7. Нажмите на кнопку-светодиод загорается/гаснет.

***Схема подключения:***

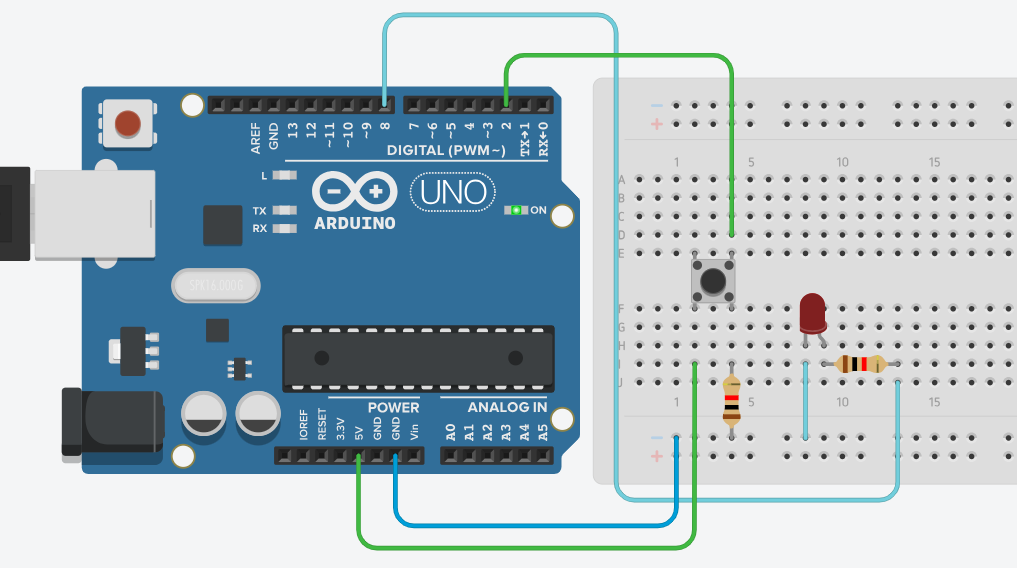
******

Рисунок 4 – Принципиальная схема подключения светодиода и тактовой кнопки

***Скетч***

int button = 2;

int led = 8;

void setup() {

pinMode(led, OUTPUT);

pinMode(button, INPUT);

}

void loop(){

if (digitalRead(button) == HIGH) {

digitalWrite(led, HIGH);

}

else {

digitalWrite(led, LOW);

}

}



**Практическая6.Семисегментныйиндикатор**

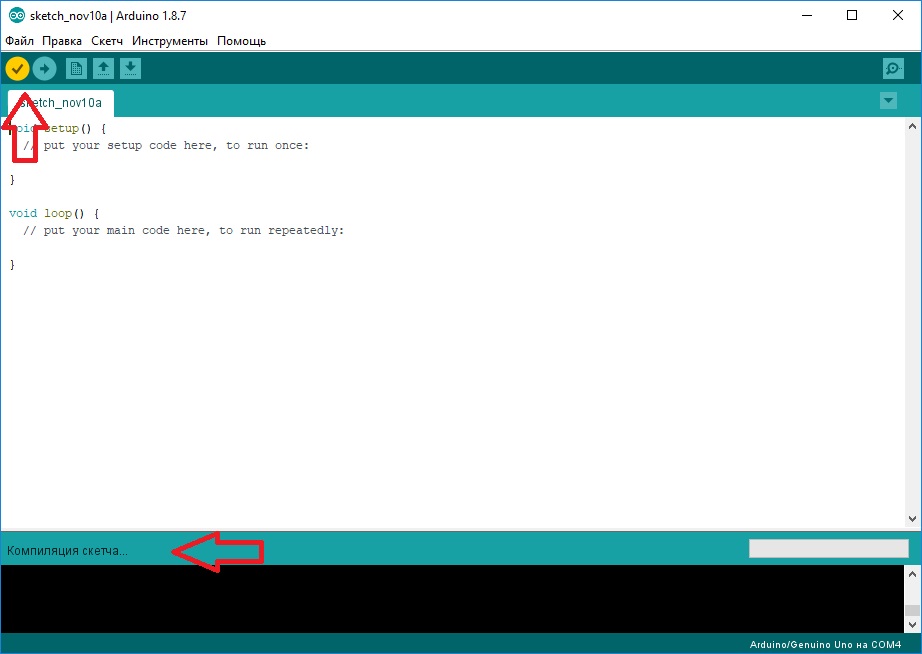
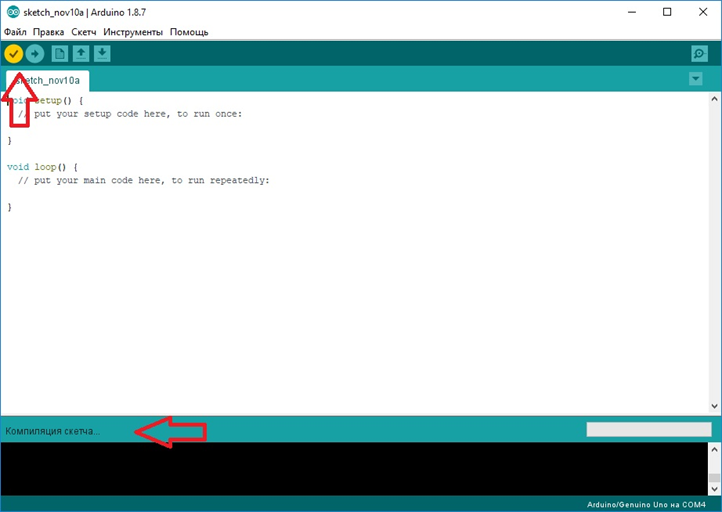
***Цель работы:***

Создание программы по отображению данных на семисегментном индикаторе.

**Оборудование**

* Микроконтроллер- Arduino UNO
* Макетная плата
* Резистор(7 шт)
* Соединительные провода «папа-папа»
* USB кабель
* Компьютер
* Семисегментный индикатор.

**Ход работы**

1. Собрать схему рисунок 5;
2. На компьютере запустить программу Arduinо;
3. Прописать программу
4. Проверить программу
5. Подключить через USB кабель
6. Загрузить программу на микрокроконтроллер
7. покрутите ручкой потенциометра – яркость светодиода должна изменятся

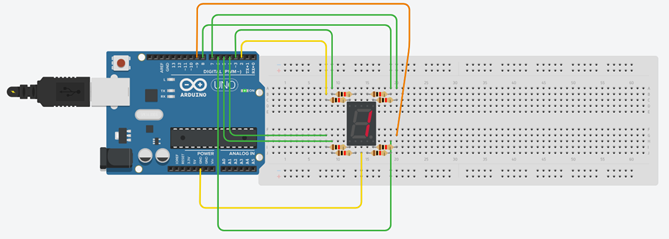
***Схема подключения:***

Рисунок 5 – Принципиальная схема подключения семисегментного индикатора

***Скетч***

#define A 8

#define B 7

#define C 6

#define D 5

#define E 4

#define F 3

#define G 2

void setup() {

pinMode(A, OUTPUT);

pinMode(B, OUTPUT);

pinMode(C, OUTPUT);

pinMode(D, OUTPUT);

pinMode(E, OUTPUT);

pinMode(F, OUTPUT);

pinMode(G, OUTPUT);

}

void loop() {

digitalWrite(A, LOW); //цифраодин

digitalWrite(B, HIGH);

digitalWrite(C, HIGH);

digitalWrite(D, LOW);

digitalWrite(E, LOW);

digitalWrite(F, LOW);

digitalWrite(G, LOW);

delay(1000);

digitalWrite(A, HIGH); //цифрадва

digitalWrite(B, HIGH);

digitalWrite(C, LOW);

digitalWrite(D, HIGH);

digitalWrite(E, HIGH);

digitalWrite(F, LOW);

digitalWrite(G, HIGH);

delay(1000);

digitalWrite(A, HIGH); //цифратри

digitalWrite(B, HIGH);

digitalWrite(C, HIGH);

digitalWrite(D, HIGH);

digitalWrite(E, LOW);

digitalWrite(F, LOW);

digitalWrite(G, HIGH);

delay(1000);

}

**Практическая 7.LCD дисплей**

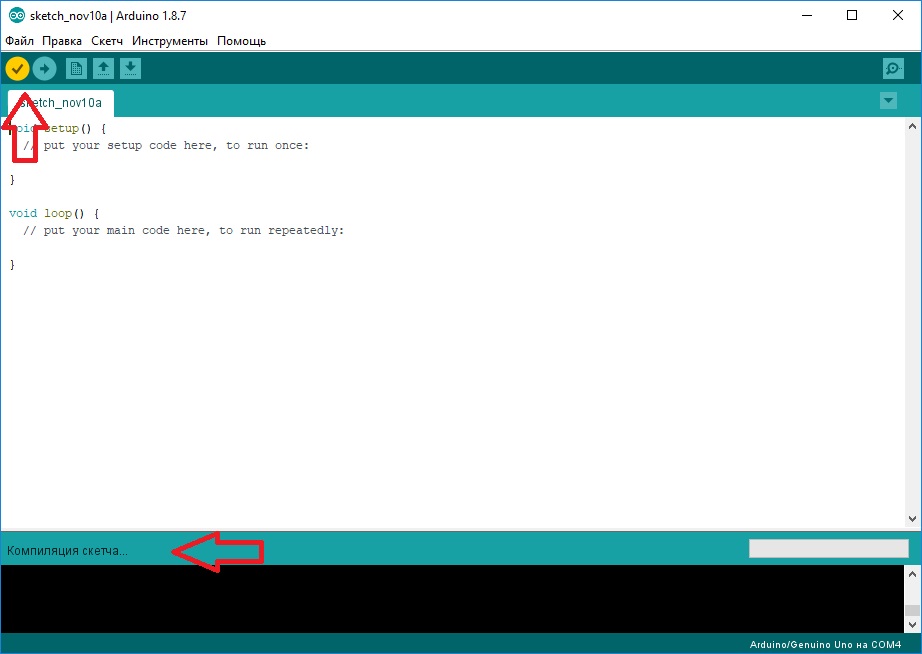
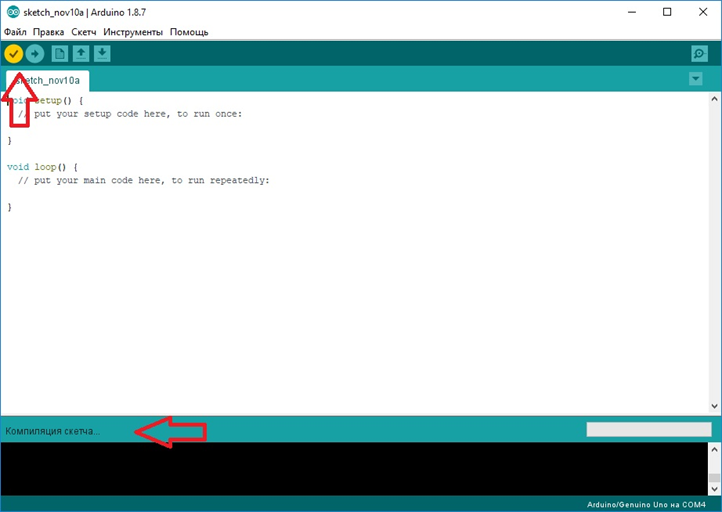
***Цель работы:***

Создание программы по выводу данных на LCD дисплей

**Оборудование**

* Микроконтроллер- Arduino UNO
* Макетная плата
* LCD дисплей
* Соединительные провода «папа-папа»,«мама-папа»
* USB кабель
* Компьютер

**Ход работы**

1. Собрать схему рисунок 6;
2. На компьютере запустить программу Arduinо;
3. Прописать программу
4. Проверить программу
5. Подключить через USB кабель
6. Загрузить программу на микрокроконтроллер

***Схема подключения:***

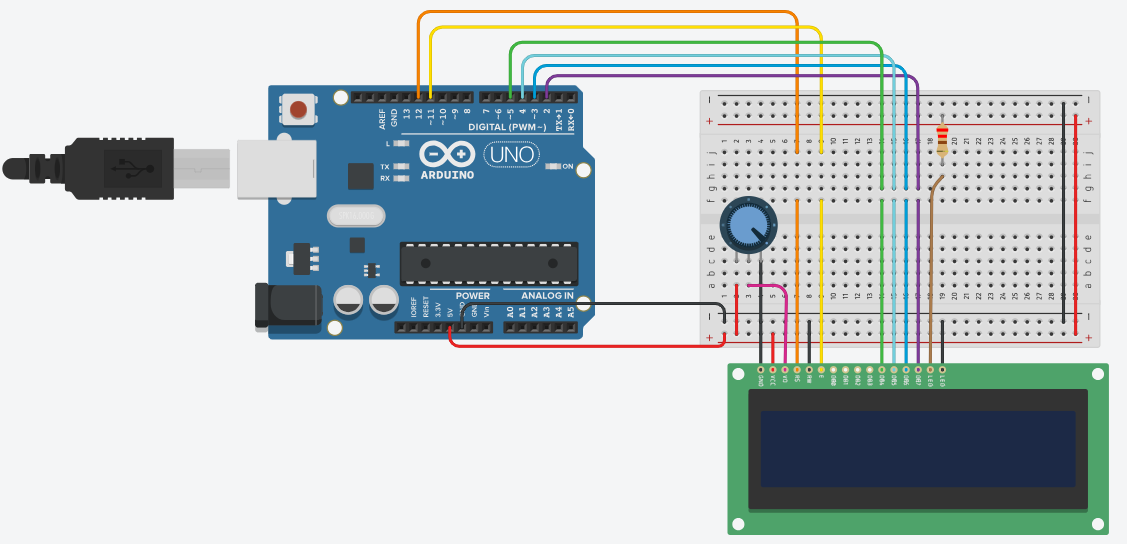
******

Рисунок 6 – Принципиальная схема подключения LCD дисплей и потенциометра

***Скетч***

// C++ code

//

/\*

LiquidCrystal Library - Hello World

Demonstrates the use of a 16x2 LCD display.

The LiquidCrystal library works with all LCD

displays that are compatible with the Hitachi

HD44780 driver. There are many of them out

there, and you can usually tell them by the

16-pin interface.

This sketch prints "Hello World!" to the LCD

and shows the time.

The circuit:

\* LCD RS pin to digital pin 12

\* LCD Enable pin to digital pin 11

\* LCD D4 pin to digital pin 5

\* LCD D5 pin to digital pin 4

\* LCD D6 pin to digital pin 3

\* LCD D7 pin to digital pin 2

\* LCD R/W pin to ground

\* LCD VSS pin to ground

\* LCD VCC pin to 5V

\* 10K resistor:

\* ends to +5V and ground

\* wiper to LCD VO pin (pin 3)

Library originally added 18 Apr 2008 by David

A. Mellis

library modified 5 Jul 2009 by Limor Fried

(http://www.ladyada.net)

example added 9 Jul 2009 by Tom Igoe

modified 22 Nov 2010 by Tom Igoe

This example code is in the public domain.

http://www.arduino.cc/en/Tutorial/LiquidCrystal

\*/

#include <LiquidCrystal.h>

int seconds = 0;

LiquidCrystal lcd\_1(12, 11, 5, 4, 3, 2);

void setup()

{

lcd\_1.begin(16, 2); // Set up the number of columns and rows on the LCD.

// Print a message to the LCD.

lcd\_1.print("hello world!");

}

void loop()

{

// set the cursor to column 0, line 1

// (note: line 1 is the second row, since counting

// begins with 0):

lcd\_1.setCursor(0, 1);

// print the number of seconds since reset:

lcd\_1.print(seconds);

delay(1000); // Wait for 1000 millisecond(s)

seconds += 1;

}



**Практическая 8.Сервопривод**

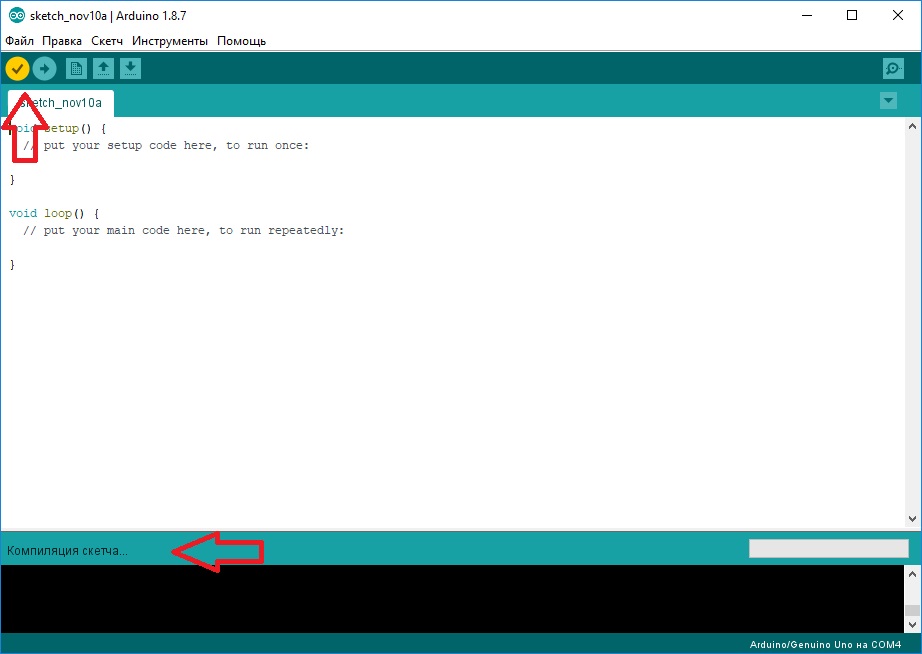
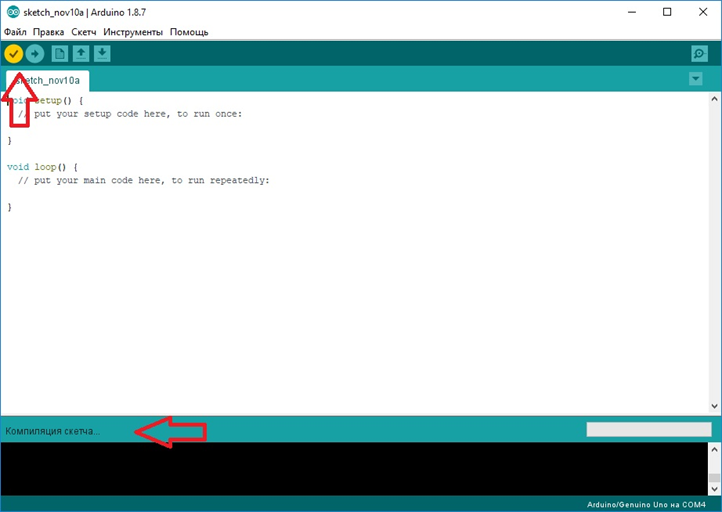
***Цель работы:***

Создание программы для управления сервоприводом.

**Оборудование**

* Микроконтроллер- Arduino UNO
* Макетная плата
* Сервопривод
* Соединительные провода «папа-папа»
* USB кабель
* Компьютер
* Питание для платы

**Ход работы**

1. Собрать схему рисунок 7;
2. На компьютере запустить программу Arduinо;
3. Прописать программу
4. Проверить программу
5. Подключить через USB кабель
6. Загрузить программу на микрокроконтроллер

***Схема подключения:***

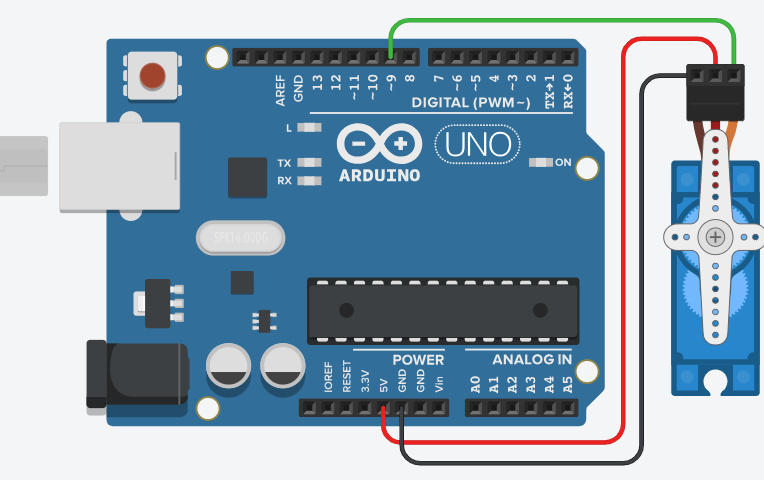
******

Рисунок 7 – Принципиальная схема подключения сервопривода

***Скетч***

// C++ code

//

/\*

Sweep

by BARRAGAN <http://barraganstudio.com>

This example code is in the public domain.

modified 8 Nov 2013 by Scott Fitzgerald

http://www.arduino.cc/en/Tutorial/Sweep

\*/

#include <Servo.h>

int pos = 0;

Servo servo\_9;

void setup()

{

servo\_9.attach(9, 500, 2500);

}

void loop()

{

// sweep the servo from 0 to 180 degrees in steps

// of 1 degrees

for (pos = 0; pos <= 180; pos += 1) {

// tell servo to go to position in variable 'pos'

servo\_9.write(pos);

// wait 15 ms for servo to reach the position

delay(15); // Wait for 15 millisecond(s)

}

for (pos = 180; pos >= 0; pos -= 1) {

// tell servo to go to position in variable 'pos'

servo\_9.write(pos);

// wait 15 ms for servo to reach the position

delay(15); // Wait for 15 millisecond(s)

}

}



э

**Практическая 9.Модуль звука с микрофоном**

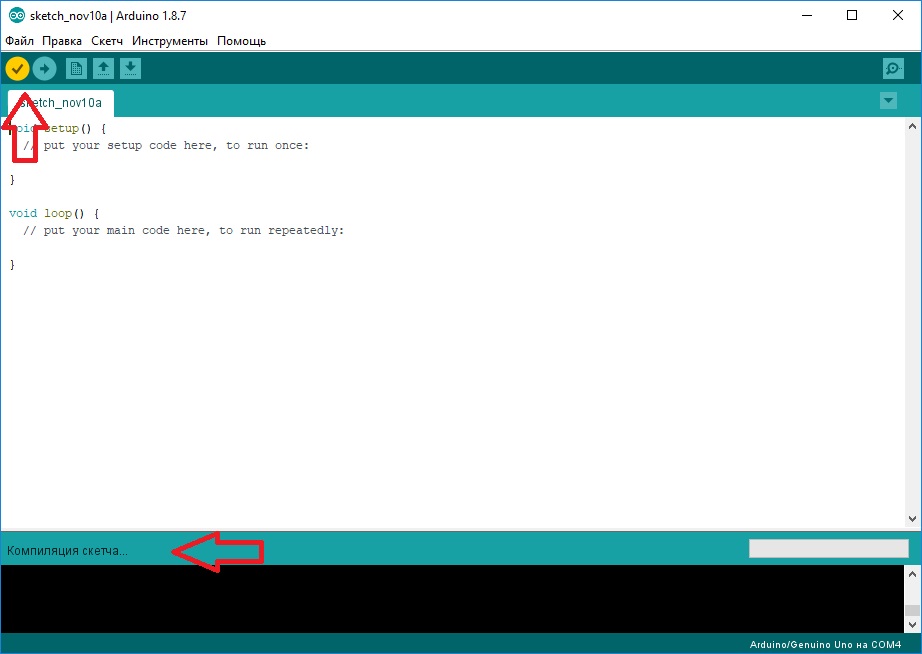
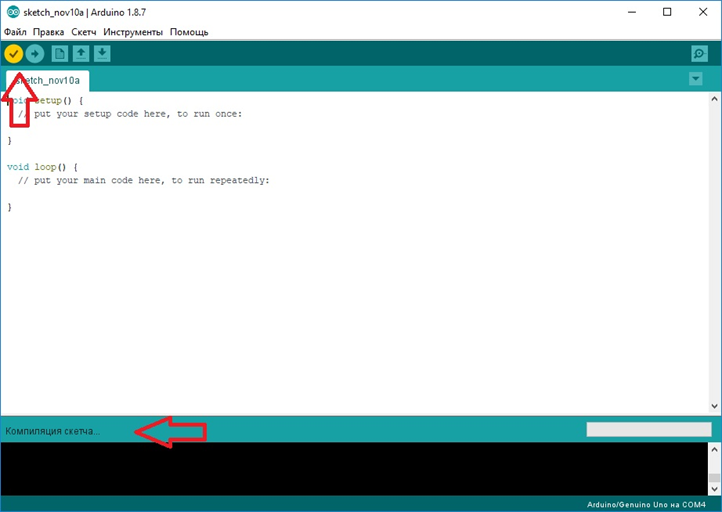
***Цель работы:***

Создание программы для управления включения светодиода по хлопку.

**Оборудование**

* Микроконтроллер- Arduino UNO
* Макетная плата
* Резистор
* Светодиод
* Соединительные провода «папа-папа»
* USB кабель
* Компьютер
* Потенциометр

**Ход работы**

1. Собрать схему рисунок 8;
2. На компьютере запустить программу Arduinо;
3. Прописать программу
4. Проверить программу
5. Подключить через USB кабель
6. Загрузить программу на микрокроконтроллер
7. Хлопнуть в ладоши-светодиод загорается.

***Схема подключения:***



Рисунок 8 – Принципиальная схема подключения модуля звука с микрофоном

***Скетч***

boolean statuslamp; // состояние лампы: true - включено, false - выключено

void setup() {

pinMode(12,OUTPUT); // пин 12 со светодиодом будет выходом (англ. «output»)

pinMode(A0,INPUT); // к аналоговому входу A0 подключим датчик (англ. «intput»)

statuslamp=false; // начальное состояние - лампа выключена

Serial.begin(9600); // подключаем монитор порта

}

void loop() {

Serial.println (analogRead(A0)); // выводим значение датчика на монитор

if(analogRead(A0)>60) {

statuslamp=!statuslamp; // меняем статус лампы при регистрации хлопка

digitalWrite(12,statuslamp); // переключаем светодиод на выходе 12

delay(20); // задержка, "дребезга" хлопков

}

****}

**Практическая 10.Светофор**

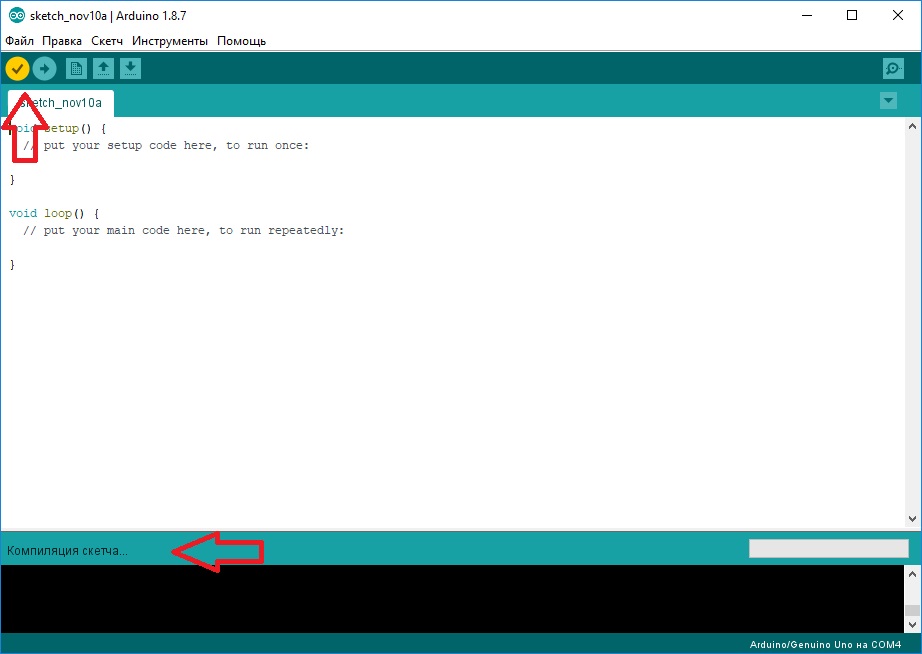
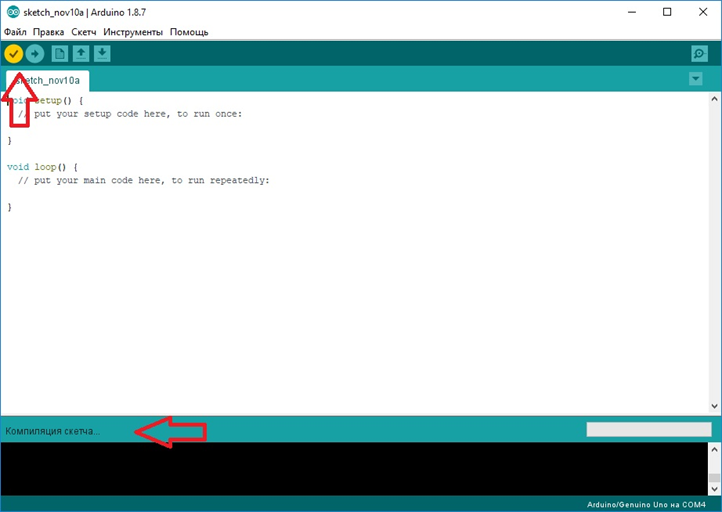
***Цель работы:***

Создание программы светофор.

**Оборудование**

* Микроконтроллер- Arduino UNO
* Макетная плата
* Резистор (3 шт)
* Светодиод(3 шт)
* Соединительные провода «папа-папа»
* USB кабель
* Компьютер

**Ход работы**

1. Собрать схему рисунок 9;
2. На компьютере запустить программу Arduinо;
3. Прописать программу
4. Проверить программу
5. Подключить через USB кабель
6. Загрузить программу на микрокроконтроллер

***Схема подключения:***

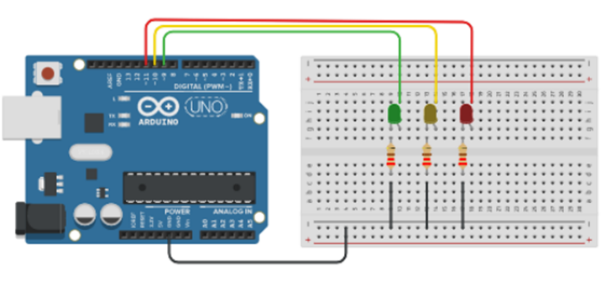
******

Рисунок 9 – Принципиальная схема подключения светодиодов

***Скетч***

const int LED\_RED = 13; // Порт 13, красныйсветодиод

const int LED\_YELLOW = 12; // Порт 12, желтыйсветодиод

const int LED\_GREEN = 11; // Порт 11, зеленыйсветодиод

const int TIMEOUT\_RED = 3000; // Время горения красного сетодиода

const int TIMEOUT\_YEL = 1690; // Время горения желтого светодиода

const int TIMEOUT\_GREEN = 2000; // Время горения зеленого светодиода

const int TIMEOUT\_FLASH\_GREEN = 500; // Времямиганиязеленогосветодиода

void setup()

{

// Все порты светодиодов будут у нас установлены в режим "внешняя нагрузка", OUTPUT

pinMode(LED\_RED, OUTPUT);

pinMode(LED\_YELLOW, OUTPUT);

pinMode(LED\_GREEN, OUTPUT);

// Устанавливаем начальное значение светодиодов

digitalWrite(LED\_RED, LOW);

digitalWrite(LED\_YELLOW, LOW);

digitalWrite(LED\_GREEN, LOW);

}

void loop()

{

// Включаемзеленыйцветсветофора

digitalWrite(LED\_GREEN, HIGH); // Включаемсветодиод

delay(TIMEOUT\_GREEN); // Ждем

// Мигаем зеленым светодиодом 3 раза

for (int i=0; i<3; i++)

{

digitalWrite(LED\_GREEN, LOW);

delay(TIMEOUT\_FLASH\_GREEN);

digitalWrite(LED\_GREEN, HIGH);

delay(TIMEOUT\_FLASH\_GREEN);

}

// Теперь отключаем зеленый и включаем желтый светодиод

digitalWrite(LED\_GREEN, LOW);

digitalWrite(LED\_YELLOW, HIGH);

delay(TIMEOUT\_YEL);

// Отключаемжелтыйсветодиод.

digitalWrite(LED\_YELLOW, LOW);

// Теперь включаем красный цвет

digitalWrite(LED\_RED, HIGH);

delay(TIMEOUT\_RED);

// Включаем желтый светодиод,не выключая красный

digitalWrite(LED\_YELLOW, HIGH);

delay(TIMEOUT\_YEL);

// Отключаем желтый и красный светодиоды.

digitalWrite(LED\_YELLOW, LOW);

digitalWrite(LED\_RED, LOW);

}

**Заключение**

В сборнике представлены практические работы с самыми простыми и часто используемыми элементами электронного конструктора Arduino, светодиод, пьезодинамик, тактовая кнопка, семисегментный индикатор, LCD дисплей,сервопривод, модуль звука с микрофоном. Практические работы в сборике размещены от более простых к сложным, что позволяе обучающимся нарабатывать навыки конструирования.

В сборнике также представленаусложнённая практическая работа «Сетофор» выполнение которой требует решения комплекса задач.

Материалы сборника могут быть использованыновичкам для самостоятельного изучения робототехнике, а также при помощи учителя