МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт математики и информационных систем

Факультет автоматики и вычислительной техники

Кафедра систем автоматизации управления

Дата сдачи на проверку:

11 февраля 2025 г.

Проверено:

11 февраля 2025 г.

**Отчет по практической работе №1**

**«Светодиодные индикаторы»**

по дисциплине

«Информационные технологии»

Вариант 1

Выполнил студент гр. УТБ-1301-02-00 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Борисов И. А/

(Подпись)

Проверил ст. преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / Шмакова Н. А/

(Подпись)

Работа защищена «\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2025 г.

Киров 2025

Цель работы: ознакомление с элементами Arduino IDE, изучение основ

работы со средой для программирования, а также сборка схем со светодиодными индикаторами.

1.1 Бегущий огонек

Индивидуальное задание представлено в таблица 1.

Таблица 1 – Индивидуальное задание

|  |  |
| --- | --- |
| Начальное состояние вкл/выкл | Движение |
| вкл | Туда/обратно с 1 |

1.2 Схема сборки бегущего огонька на макетной плате представлена на рисунке 1.1

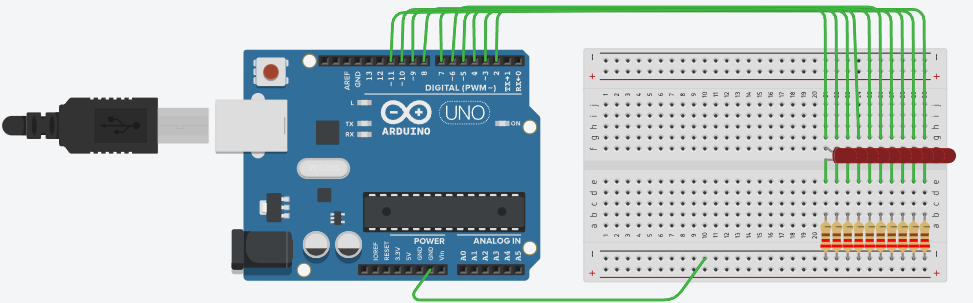


Рисунок 1.1 – Схема сборки на макетной плате

1.3 Принципиальная схема представлена на рисунке 1.2

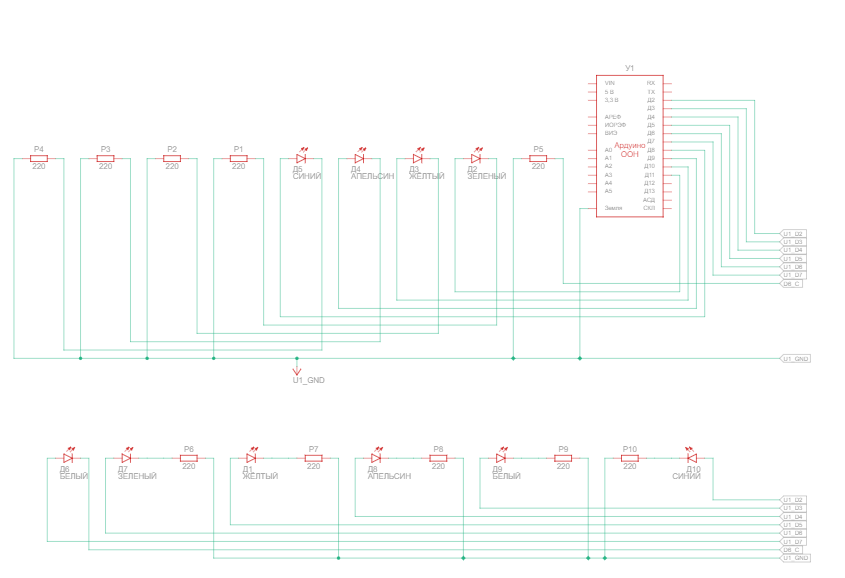


Рисунок 1.2 – Принципиальная схема

1.4 Листинг программного кода

void setup() {  
 for (int pin = 1; pin <= 11; ++pin)  
 pinMode(pin, OUTPUT);  
}  
  
void loop() {  
 unsigned int ms = millis(); *// значение времени в миллисекундах* int pin = 1 + 1+(ms / 120/2)%20; *// значение пин меняется каждые 120 миллисекунд* if (pin <= 11) {  
 digitalWrite(pin, HIGH);  
 delay(10);  
 digitalWrite(pin, LOW);  
 } else {  
 int reversePin = 23 - pin;  
 digitalWrite(reversePin, HIGH);  
 delay(10);  
 digitalWrite(reversePin, LOW);  
 }  
}

1.5 Ссылка на рабочий проект

[Бегущий огонек](https://www.tinkercad.com/things/3cExjQ2Z561-3-baza-begushij-ogonek/editel?returnTo=https%3A%2F%2Fwww.tinkercad.com%2Fdashboard%2Fdesigns%2Fcircuits%3Fpage%3D2)

2.1 Пульсар

Индивидуальное задание представлено в таблице 2.

Таблица 2 – Индивидуальное задание.

|  |  |  |
| --- | --- | --- |
| Начальное состояние  вкл/выкл | Транзистор | Смена состояний |
| вкл | биполярный | Увеличение до max/уменьшение min |

2.2 Схема сборки пульсара на макетной плате представлена на рисунке 2.1

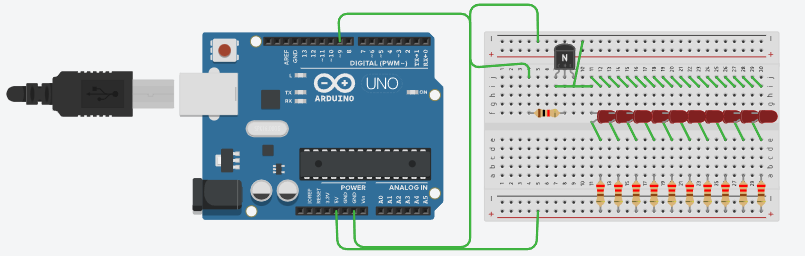


Рисунок 2.1 – Схема сборки на макетной плате

2.3 Принципиальная схема представлена на рисунке 2.2

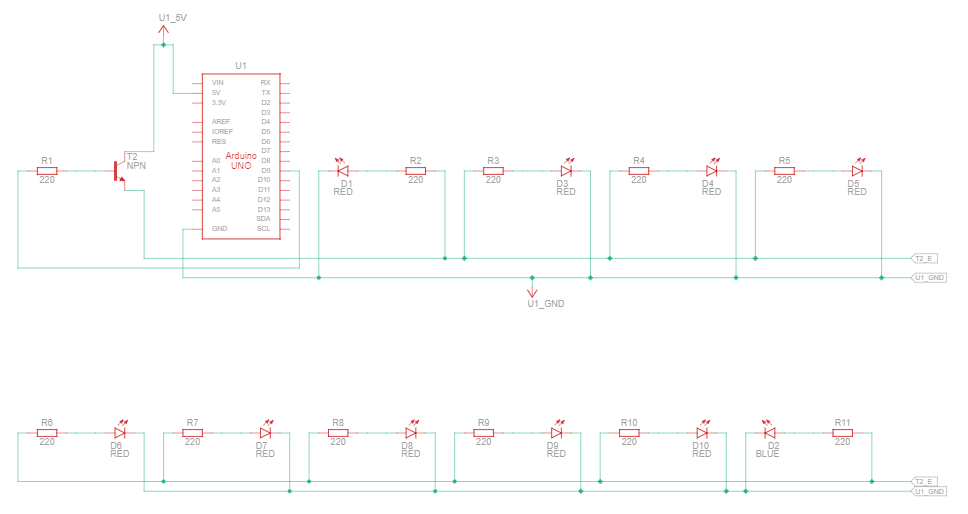


Рисунок 2.2 – Принципиальная схема

2.4 Листинг программного кода.

const int ledPin = 9;  
const int minBrightness = 0;  
const int maxBrightness = 255;  
const int fadeDelay = 10; *//задержка*const int fadeStep = 1; *//величина, на которую изменяется яркость на каждой итерации*int brightness = minBrightness;  
int fadeDirection = 1;  
  
void setup() {  
 Serial.begin(9600);  
 pinMode(ledPin, OUTPUT);  
}  
  
void loop() {  
 analogWrite(ledPin, brightness); *//используется для установки яркости светодиода в значение brightness.* brightness += fadeDirection \* fadeStep;  
  
 if (brightness <= minBrightness || brightness >= maxBrightness) {  
 fadeDirection \*= -1; *//изменение яркости в обратном направлении* }  
 Serial.println(brightness);  
 delay(fadeDelay);  
}

2.5 Ссылка на рабочий проект

[Пульсар](https://www.tinkercad.com/things/eWysVTjLHKK-pulsar/editel?returnTo=https%3A%2F%2Fwww.tinkercad.com%2Fdashboard%2Fdesigns%2Fcircuits%3Fpage%3D1)

3.1 Ночной светильник

Индивидуальное задание представлено в таблице 3

Таблица 3 – Индивидуальное задание.

|  |  |  |  |
| --- | --- | --- | --- |
| Начальное состояние вкл/выкл | Входной сигнал | Делитель напряжения | Смена состояние |
| выкл | фоторезистер | 1 кОм | Увеличение в зависимости от  значения на датчике |

3.2 Схема сборки пульсара на макетной плате представлена на рисунке 3.1

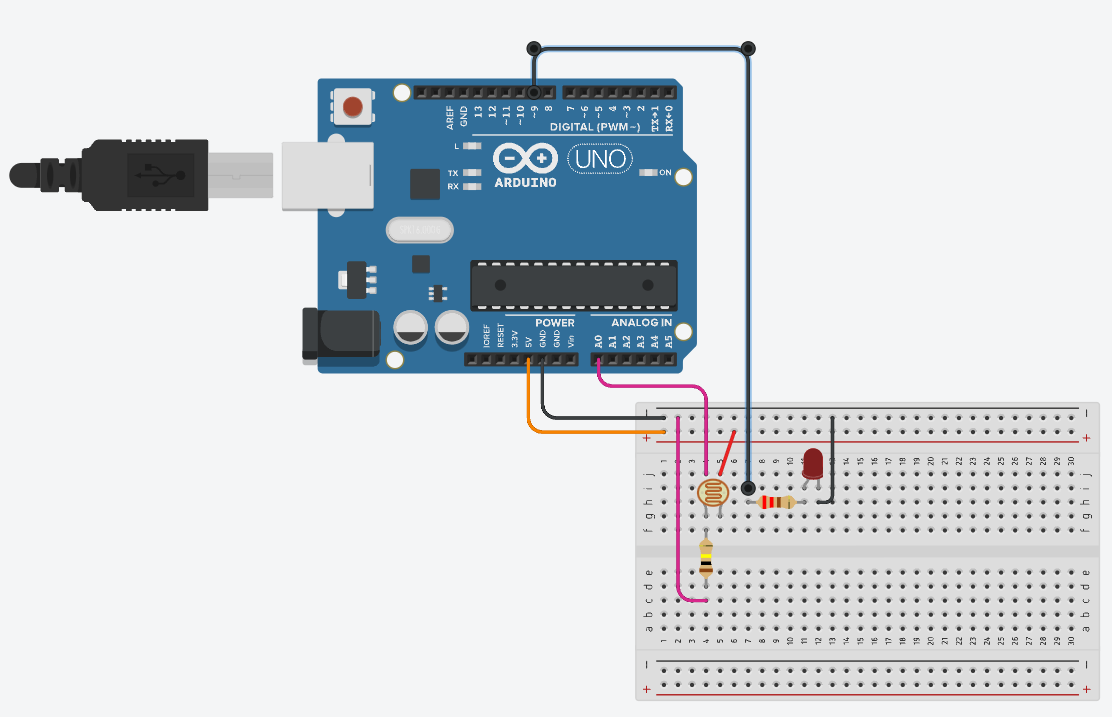


Рисунок 3.1 – Принципиальная схема

3.3 Принципиальная схема представлена на рисунке 3.2

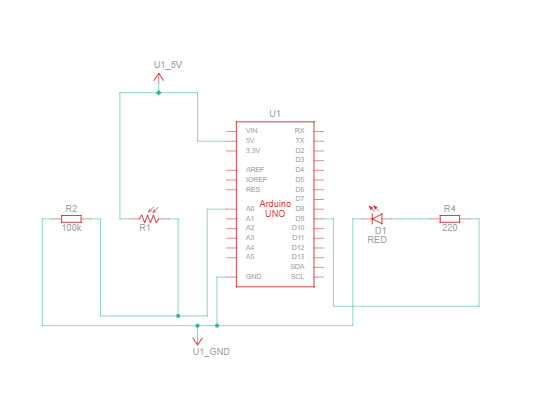


Рисунок 3.2 – Принципиальная схема

3.4 Листинг программного кода

#define LED\_PIN 9  
 #define LDR\_PIN A0 *//имя для пина с фоторезистором*void setup()  
{  
 pinMode(LED\_PIN, OUTPUT);  
 Serial.begin(9600);  
}  
void loop()  
{  
  
 int lightness = analogRead(LDR\_PIN); *//считываем уровень освещенности* Serial.println(lightness);  
 analogWrite(LED\_PIN, map(lightness, 366, 1018, 0, 255));  
}

3.5 Ссылка на рабочий проект

[Ночной светильник](https://www.tinkercad.com/things/gz93hcBXXSI-12-gotovyj-pulsarlaba/editel?returnTo=%2Fdashboard%3Fcollection%3Ddesigns%26type%3Dcircuits&sharecode=YPNIBIJ__LpCQVb0oLUD-6ZQO1LORU-ZlU4ut3zr4pU)

4.1 Кнопочный переключатель

Есть 2 кнопки и 1 светодиод. Необходимо включать и выключать светодиод в соответствии с заданием, указанным в таблице. Индивидуальное задание представлено в таблице 4.

Таблица 4 – Индивидуальное задание

|  |  |  |  |
| --- | --- | --- | --- |
| Включение | Направление | Выключение | Направление |
| 2 | И слева и справа | 2 | Только слева |

4.2 Схема сборки на макетной плате представлена на рисунке 4.1

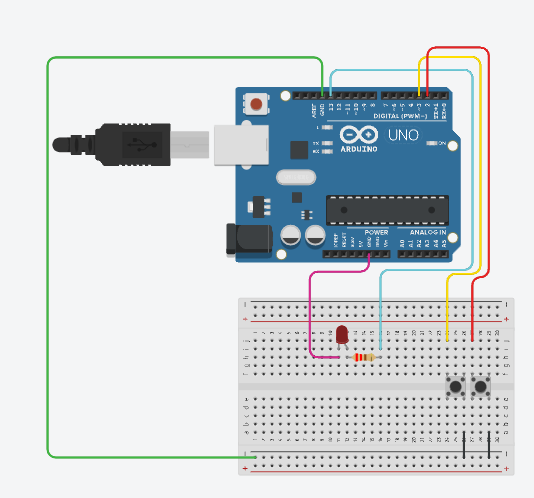


Рисунок 4.1 – Схема сборки на макетной плате

4.3 Принципиальная схема представлена на рисунке 4.2

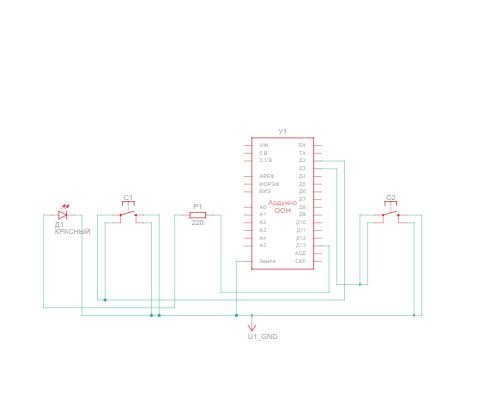


Рисунок 4.2 – Принципиальная схема

4.4 Листинг программного кода

#define RIGHT\_BUTTON\_PIN 2  
 #define LEFT\_BUTTON\_PIN 3  
 #define LED\_PIN 13  
boolean firstLeftPressed = false;  
boolean secondRightPressed = false;  
boolean flag = false;  
boolean ledEnabled = false; *// включён ли свет?*void setup()  
{  
 Serial.begin(9600);  
 pinMode(LED\_PIN, OUTPUT);  
 pinMode(LEFT\_BUTTON\_PIN, INPUT\_PULLUP);  
 pinMode(RIGHT\_BUTTON\_PIN, INPUT\_PULLUP);  
}  
  
void loop()  
{  
 if(!ledEnabled)*//велючение* {  
 firstLeftPressed = digitalRead(LEFT\_BUTTON\_PIN);  
 if(!firstLeftPressed)  
 {  
 digitalWrite(LED\_PIN, HIGH);  
 ledEnabled = true;  
 delay(1000);  
 }  
 }  
 else*//выключение* {  
 firstLeftPressed = digitalRead(LEFT\_BUTTON\_PIN);  
 secondRightPressed = digitalRead(RIGHT\_BUTTON\_PIN);  
 Serial.println(!firstLeftPressed);  
 if (!firstLeftPressed and secondRightPressed) flag = true;  
 if (!secondRightPressed and flag)  
 {  
 firstLeftPressed = false;  
 secondRightPressed = false;  
 flag = false;  
 ledEnabled = false;  
 digitalWrite(LED\_PIN, LOW);  
 }  
 }  
}

4.5 Ссылка на рабочий проект

[Кнопочный переключатель](https://www.tinkercad.com/things/eqKYtoBrIVu-14-knopki-labagotovaya/editel?returnTo=%2Fdashboard%3Fcollection%3Ddesigns%26type%3Dcircuits&sharecode=jjQmZYdzFtdtb4Bb46Xp5zjjc2CiVjxprBSW55etWF0)

5.1 RGB светодиод

Индивидуальное задание представлено в таблице 5.

Таблица 5 – Индивидуальное задание

|  |  |  |  |
| --- | --- | --- | --- |
| фиксация | Срабатывание  (нажатие/отпускание) | Смешивание цветов | Светодиод с общим анодом/катодом |
| вкл | нажатие | выкл | катод |

5.2 Схема сборки на макетной плате представлена на рисунке 5.1

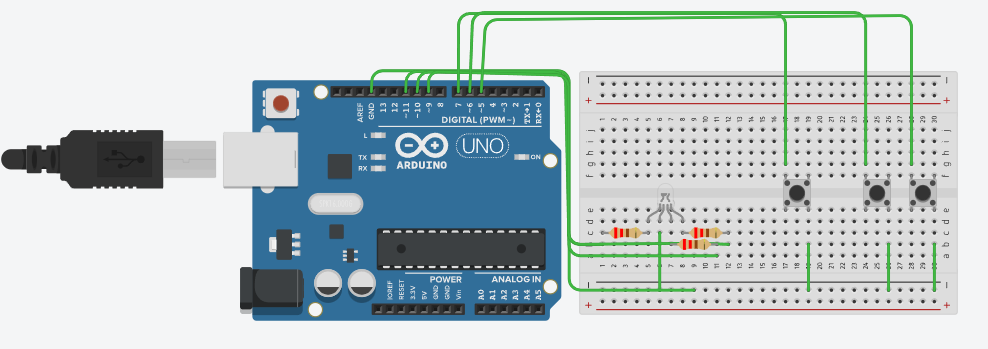


Рисунок 5.1 – Схема сборки на макетной плате

5.3 Принципиальная схема представлена на рисунке 4.3

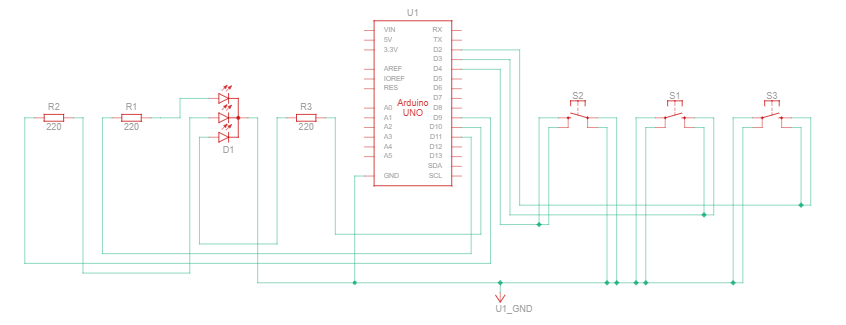


Рисунок 5.1 – Принципиальная схема

*// C++ code*int buttonsPins[] = {5, 6, 7};  
int*// C++ code*int buttonsPins[] = {5, 6, 7};  
int ledsPins[] = {9, 10, 11};  
int Statebutton[] = {1, 1, 1};  
int previousState[] = {1, 1, 1};  
int brightness = 0;  
int k = 0;  
int colors[] = {LOW, LOW, LOW};  
void setup() {  
 for (int i = 0; i < 3; i++) {  
 pinMode(ledsPins[i], OUTPUT);  
 pinMode(buttonsPins[i], INPUT\_PULLUP);  
 }  
}  
void loop() {  
 Statebutton[k] = digitalRead(buttonsPins[k]);  
 if (Statebutton[k] != previousState[k]) {  
 if (Statebutton[k] == LOW) {  
 colors[k] = (colors[k] == LOW) ? HIGH : LOW;  
 digitalWrite(ledsPins[k], colors[k]);  
 int i = (k + 1) % 3;  
 if (i != k) {  
 colors[i] = LOW;  
 digitalWrite(ledsPins[i], colors[i]);  
 }  
 i = (k + 2) % 3;  
 if (i != k) {  
 colors[i] = LOW;  
 digitalWrite(ledsPins[i], colors[i]);  
 }  
 }  
 }  
 previousState[k] = Statebutton[k];  
 k = (k + 1) % 3;  
} ledsPins[] = {9, 10, 11};  
int Statebutton[] = {1, 1, 1};  
int previousState[] = {1, 1, 1};  
int brightness = 0;  
int k = 0;  
int colors[] = {LOW, LOW, LOW};  
void setup() {  
 for (int i = 0; i < 3; i++) {  
 pinMode(ledsPins[i], OUTPUT);  
 pinMode(buttonsPins[i], INPUT\_PULLUP);  
 }  
}  
void loop() {  
 Statebutton[k] = digitalRead(buttonsPins[k]);  
 if (Statebutton[k] != previousState[k]) {  
 if (Statebutton[k] == LOW) {  
 colors[k] = (colors[k] == LOW) ? HIGH : LOW;  
 digitalWrite(ledsPins[k], colors[k]);  
 int i = (k + 1) % 3;  
 if (i != k) {  
 colors[i] = LOW;  
 digitalWrite(ledsPins[i], colors[i]);  
 }  
 i = (k + 2) % 3;  
 if (i != k) {  
 colors[i] = LOW;  
 digitalWrite(ledsPins[i], colors[i]);  
 }  
 }  
 }  
 previousState[k] = Statebutton[k];  
 k = (k + 1) % 3;  
}

5.5 Ссылка на рабочий проект.

[RGB светодиод](https://www.tinkercad.com/things/lnMb6VhqzeG-rgb-9/editel?returnTo=https%3A%2F%2Fwww.tinkercad.com%2Fdashboard%2Fdesigns%2Fcircuits%3Fpage%3D1)

Вывод

В ходе данной работы ознакомились с элементами Arduino IDE, посмотрели основы работы с этой средой программирования, а также освоил процесс сборки схем с использованием светодиодных индикаторов.