Федеральное государственное бюджетное образовательное учреждение высшего

образования

«Чувашский государственный университет им. И.Н. Ульянова»

Кафедра вычислительной техники

Лабораторная работа №2

**Управление светодиодом и кнопкой с использованием Arduino Nano.**

Выполнил: Иванов В.С.

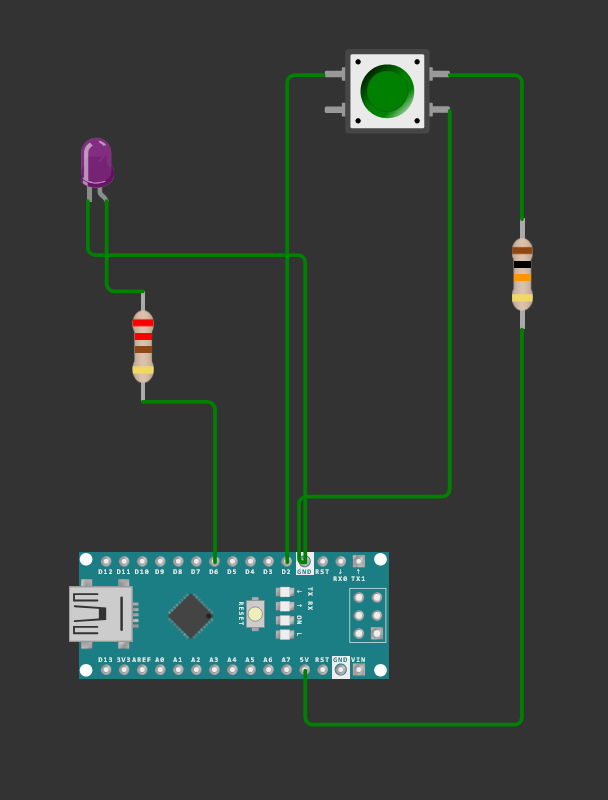
студент группы ИВТ-41-22

Проверил: Степанов Сергей Владимирович

Чебоксары, 2025

Цель работы: Изучить работу с цифровыми и аналоговыми выходами Arduino Nano, освоить использование кнопки для управления светодиодом, а также работу с ШИМ (PWM) для изменения яркости светодиода.

1 Задание:



Код программы:

int brightness = 0; // уставливаем начально значение яркости

bool button;

bool lastButton = HIGH;

void setup() {

// put your setup code here, to run once:

pinMode(2, INPUT\_PULLUP);

pinMode(6, OUTPUT);

}

void loop() {

// put your main code here, to run repeatedly:

button = digitalRead(2);

if (button != lastButton) {

delay(20);

lastButton = button;

}

if (button == LOW) {

if (brightness < 255) brightness += 1;

} else {

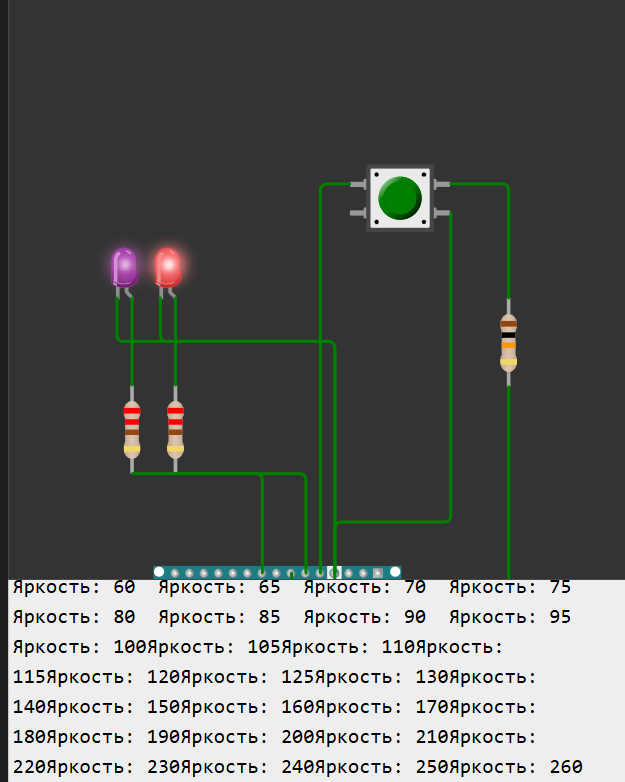
if (brightness > 0) brightness -= 1;

}

analogWrite(6, brightness);

delay(10);

}

Задание 2:  


Код программы:

int brightness = 0;

bool button;

bool lastButton = HIGH;

int prevBrightness = -1;

int changePos = 1;

void setup() {

// put your setup code here, to run once:

pinMode(2, INPUT\_PULLUP);

pinMode(6, OUTPUT);

pinMode(3, OUTPUT);

Serial.begin(9600);

Serial.print("Яркость: ");

}

void loop() {

// put your main code here, to run repeatedly:

button = digitalRead(2);

if (button != lastButton) {

delay(20);

lastButton = button;

}

int brightStep = 5;

if (brightness > 128) {

brightStep = 10;

digitalWrite(3, HIGH);

} else {

digitalWrite(3, LOW);

}

if (button == LOW) {

if (brightness < 255) brightness += brightStep;

} else {

if (brightness > 0) brightness -= brightStep;

}

analogWrite(6, brightness);

if (brightness != prevBrightness) {

Serial.print("\rЯркость: ");

Serial.print(brightness);

if (brightness < 10) Serial.print(" ");

else if (brightness < 100) Serial.print(" ");

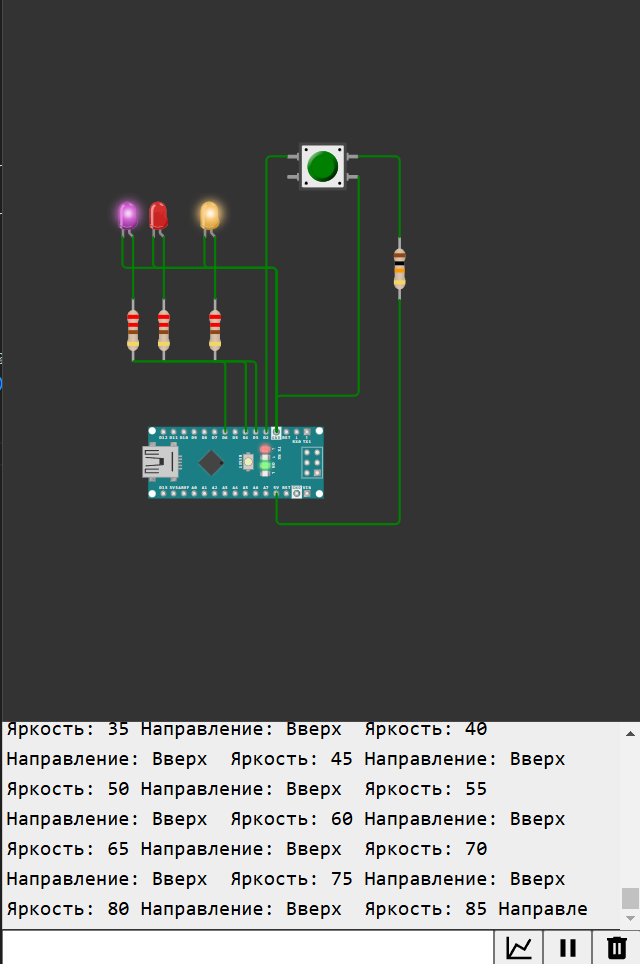
prevBrightness = brightness;

} else {

delay(10);

}

}  
  
  
Дополнительное задание:



Код программы:

int brightness = 0;

bool button;

bool lastButton = HIGH;

int prevBrightness = -1;

int changePos = 1;

int direction = 1;

unsigned long lastButtonPressTime = 0;

const int doubleClickDelay = 250;

int led3State = LOW;

unsigned long previousMillis = 0;

void setup() {

// put your setup code here, to run once:

pinMode(2, INPUT\_PULLUP);

pinMode(6, OUTPUT);

pinMode(3, OUTPUT);

pinMode(4, OUTPUT);

Serial.begin(9600);

Serial.print("Яркость: ");

}

void loop() {

// put your main code here, to run repeatedly:

unsigned long currentMillis = millis();

button = digitalRead(2);

if (button == LOW && lastButton == HIGH) {

if (currentMillis - lastButtonPressTime < doubleClickDelay) {

direction \*= -1; // Изменяем направление

Serial.println("\nНаправление изменено!");

}

lastButtonPressTime = currentMillis;

}

if (button != lastButton) {

delay(20);

lastButton = button;

}

int brightStep = 5;

if (brightness > 128) {

brightStep = 10;

digitalWrite(3, HIGH);

} else {

digitalWrite(3, LOW);

}

if (button == LOW) {

brightness = constrain(brightness + brightStep \* direction, 0, 255);

} else {

brightness = constrain(brightness - brightStep \* direction, 0, 255);

}

analogWrite(6, brightness);

if (brightness > 0) {

int blinkInterval = map(brightness, 0, 255, 1000, 50); // Частота от 1 Гц до 20 Гц

if (currentMillis - previousMillis >= blinkInterval) {

previousMillis = currentMillis;

led3State = !led3State;

digitalWrite(4, led3State);

}

} else {

digitalWrite(4, LOW);

}

if (brightness != prevBrightness) {

Serial.print("\rЯркость: ");

Serial.print(brightness);

Serial.print(" Направление: ");

Serial.print(direction == 1 ? "Вверх" : "Вниз");

if (brightness < 10) Serial.print(" ");

else if (brightness < 100) Serial.print(" ");

prevBrightness = brightness;

} else {

delay(10);

}

}

Ответы на вопросы:

1. Как работает ШИМ (PWM) в Arduino?

ШИМ (PWM — Pulse Width Modulation) — это метод управления средней мощностью, подаваемой на нагрузку (например, светодиод), путем быстрого включения и выключения сигнала.

Принцип работы:

* Частота ШИМ (обычно ~490 Гц или ~980 Гц в Arduino) определяет, как быстро сигнал включается/выключается.
* Скважность (Duty Cycle) — это процент времени, когда сигнал находится в высоком состоянии (HIGH). Например:
  + 0% — сигнал всегда выключен (0 В).
  + 50% — половину времени HIGH, половину LOW.
  + 100% — сигнал всегда включен (5 В).

В Arduino ШИМ реализуется через analogWrite(pin, value), где:

* pin — только пины с ~ (3, 5, 6, 9, 10, 11 на Uno).
* value — число от 0 (0%) до 255 (100%).

1. Почему важно использовать подтягивающий резистор для кнопки?

При подключении кнопки к Arduino без подтягивающего резистора входной пин может находиться в "плавающем" (неопределенном) состоянии, когда кнопка не нажата. Это приводит к ложным срабатываниям из-за наводок.

1. Как можно изменить скорость изменения яркости без использования второго светодиода?

Вместо второго светодиода можно попробовать использовать потенциометр, кнопку, и возможно последовательный порт.