**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В. И. Ульянова (Ленина)**

**Кафедра информационной безопасности**

отчет

**по лабораторной работе №1**

**по дисциплине «Проектирование цифровых устройств»**

Тема: Платформа Arduino: загрузка скетчей, цифровой ввод-вывод

|  |  |  |
| --- | --- | --- |
| Студент гр. 3363 |  | Минко Д. А. |
| Студент гр. 3363 |  | Гончаренко О.Д. |
| Студент гр. 3363 |  | Овсейчик Н.И. |
| Преподаватель |  | Рыбин В. Г. |

**Цель работы**

Ознакомление с основами разработки цифровых устройств на платформе Arduino путем создания простейшего функционирующего устройства на плате Arduino UNO.

**Исходное задание**

«Налобный фонарик». Имитация режимов налобных фонариков. Чтобы включить или выключить фонарик, необходимо удерживать кнопку нажатой более 1 с. Нажатие на кнопку в режиме горения переключает режим непрерывного горения и мигания.

**Ход выполнения работы**

В программе “Tincercad” собрана модель рабочего устройства (рис.1).

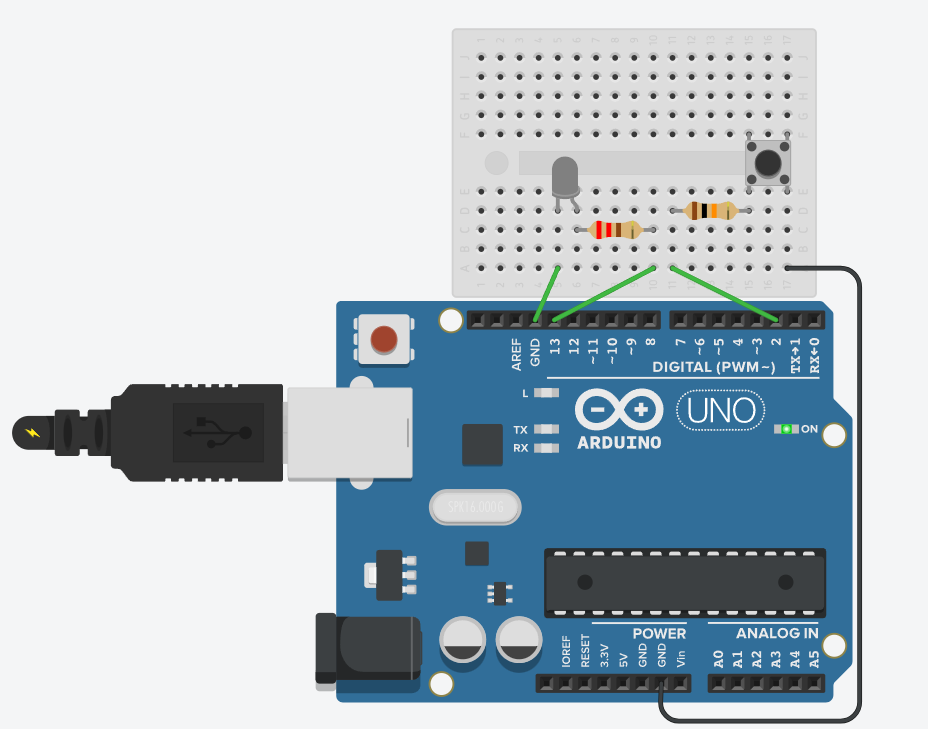


Рисунок 1 – Принципиальная схема устройства

Ниже представлен список используемых компонентов (рис.2), где резистор “R2” – резистор для кнопки, а резистор “R3” – резистор для светодиода.

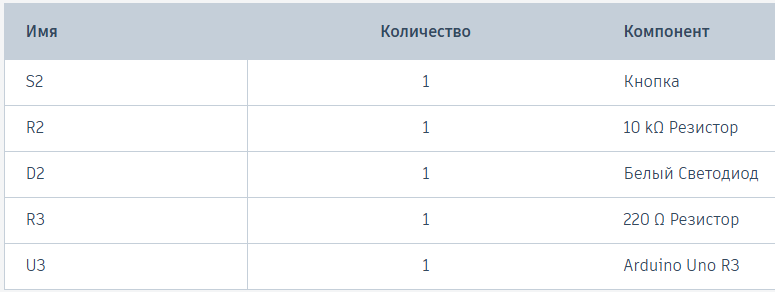
****

Рисунок 2 – Список используемых компонентов

Изображение выглядит как Электронная техника, электроника, Компонент схемы, Электронный компонент

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 3 – Прототип на макетной плате

**Принцип работы цифрового устройства**

При первоначально выключенном светодиоде его непрерывное горение активируется удержанием кнопки в течение более чем одной секунды. Короткие нажатия (менее одной секунды) переключают режим работы между постоянным свечением и миганием. Для полного отключения светодиода необходимо снова удерживать кнопку не менее одной секунды.

Кнопка подключена через резистор сопротивлением 10 кОм, её вторая ножка заземлена. Анод светодиода получает питание, а катод соединён с землёй через резистор 220 Ом, что обеспечивает его корректную работу.

**Вывод**

В ходе выполнения лабораторной работы была разработана и реализована модель налобного фонарика на платформе Arduino UNO. Устройство успешно имитирует режимы работы фонарика, включая включение и выключение по длительному нажатию кнопки, а также переключение между режимами непрерывного горения и мигания при коротком нажатии.

Принципиальная схема была собрана в среде Tinkercad, а затем реализована на макетной плате. Работа устройства подтверждена экспериментально. Использование внешнего подтягивающего резистора позволило корректно обрабатывать нажатия кнопки.

В результате работы были приобретены практические навыки проектирования цифровых устройств, работы с кнопками и светодиодами, а также программирования логики их работы на платформе Arduino.

Изображение выглядит как текст, диаграмма, Прямоугольник, линия

Контент, сгенерированный ИИ, может содержать ошибки.

ИСХОДНЫЙ КОД

Исходный код программы:

#define LED\_PIN 13 // вывод управления красным светодиодом

#define BUTTON\_PIN 2 // вывод подключения кнопки

bool buttonPressed = false; // Флаг, указывающий, что кнопка нажата

bool ledLight = false; // Флаг, показывающий, горит ли лампа

unsigned long pressStartTime = 0; // Время начала нажатия кнопки

int tmp;

void setup() {

  pinMode(LED\_PIN, OUTPUT);

  //pinMode(BUTTON\_PIN, INPUT\_PULLUP); // Используем встроенный подтягивающий резистор

  pinMode(BUTTON\_PIN, INPUT);

  digitalWrite(BUTTON\_PIN, HIGH);

}

void flashing() {

  digitalWrite(LED\_PIN, LOW);

  delay(400);

  digitalWrite(LED\_PIN, HIGH);

  delay(400);

}

void work(int arg) {

  switch (arg) {

    case 0:

      digitalWrite(LED\_PIN, LOW);

      break;

    case 1:

      digitalWrite(LED\_PIN, HIGH);

      break;

    case 2:

      flashing();

      break;

    default:

      break;

  }

}

void loop() {

  int buttonState = digitalRead(BUTTON\_PIN); // Читаем состояние кнопки

  if (buttonState == LOW) {

    if (!buttonPressed) { // Если кнопка только что нажата

      buttonPressed = true; // Устанавливаем флаг

      pressStartTime = millis(); // Запоминаем время начала нажатия

    }

    // Если кнопка удерживается более 1 секунд

    if (millis() - pressStartTime >= 1000) {

      if (ledLight == false) {

        tmp = 1;

        work(tmp);

        delay(600);

        ledLight = true;

      } else {

        tmp = 0;

        work(tmp);

        delay(600);

        ledLight = false;

      }

      pressStartTime = 0; // Сброс времени

      buttonPressed = false; // Сбрасываем флаг нажатия

    }

  } else {

    if (millis() - pressStartTime <= 1000){

      if (tmp == 2) tmp = 1;

      else if (tmp == 1) tmp = 2;

      pressStartTime = 0;

    }

    buttonPressed = false; // Сбрасываем флаг, если кнопка не нажата

  }

  work(tmp);

}