

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра САПР

ОТЧЕТ по лабораторной работе №2
по дисциплине «Проектирование цифровых устройств»
Тема: Управление временем на Arduino

Студент гр. 3364

Юсфи_А.

Преподаватель

Михайлов А.А

Санкт-Петербур

2025

Цель работы.

Реализовать гирлянду, управление временем в которой производится (переключение режимов по кнопке) альтернативными способами: delay, millis, timer (прерывания).

Ход работы

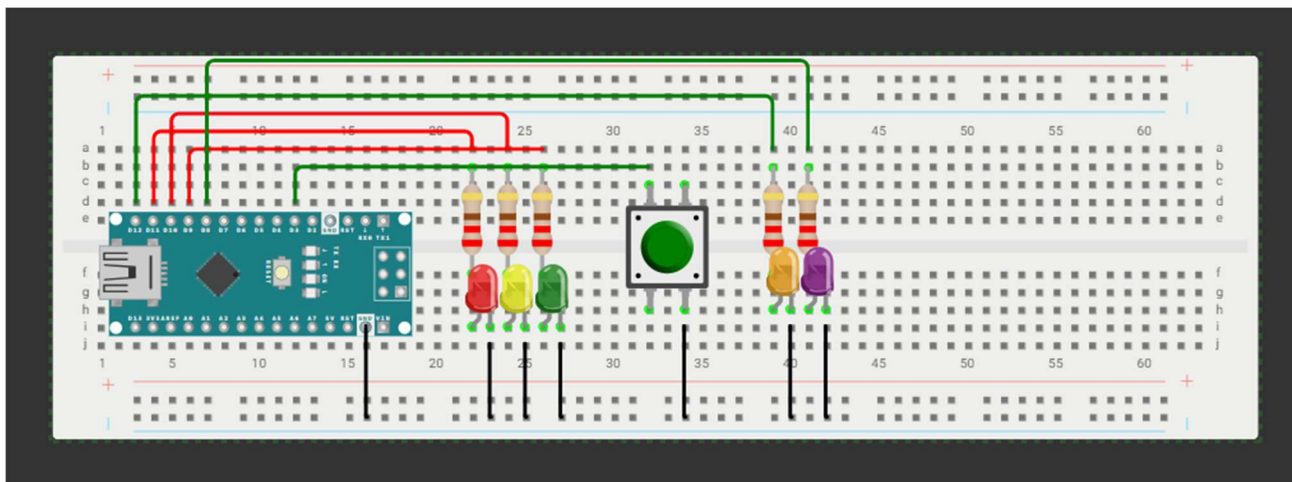


Рисунок 1 – Принципиальная схема устройства.

Рисунок 2 – Прототип на макетной плате.

Принцип работы устройства

Устройство имеет 4 режима работы:

1. Все светодиоды выключены.
2. Гирлянда с помощью delay.
3. Гирлянда с помощью millis.
4. Гирлянда с помощью timer (прерываний).

При включении устройства устанавливается режим 1. Каждое нажатие на кнопку переключает режим на следующий. После 4 режима идёт режим 1 и так по кругу.

Режим работы 2 включает гирлянду, где все светодиоды поочерёдно моргают. Режим работы 3 включает гирлянду, где первые два светодиода моргают

поочерёдно, а третий моргает быстро. Режим работы 4 включает гирлянду, где сначала загорается первый светодиод, потом третий и второй.

Гаснут они от третьего к первому.

К светодиодам подключены резисторы с сопротивлением в 220 Ом, общий катод подключен к земле. Используется кнопка, которая подключена к земле через подтягивающий резистор Arduino.

Вывод

В результате выполнения работы мы научились управлять временем на Arduino. Работа показала, что `delay` прост и удобен, но «тормозит» всю работу программы, `millis` крайне удобен и не блокирует программу, но немного более сложен в реализации чем `delay`, `timer` (прерывания) используется для немедленного выполнения какой-то заданной программы вне основной.

Листинг кода

```
#include <TimerOne.h> //Библиотека для прерываний
```

```
const int Led1 = 12; //Блок подключений
```

```
const int Led2 = 11;
```

```
const int Led3 = 10;
```

```
const int Led4 = 9;
```

```
const int Led5 = 8;
```

```
const int ButtonPin = 3;
```

```
bool flag = 0; //Флаги и переключатели
```

```
bool Button;
```

```
int Counter = 0;
```

```
int Led_pos = 0;
```

```
bool Led_pos_3 = 0;
```

```
int Led_pos_timer = 0;
```

```
volatile bool Led_Zero = false;
```

```

unsigned long last_time = 0; //Управление временем
для Millis()
unsigned long led_time = 0;
unsigned long led_time_3 = 0;

void setup() {
  pinMode(Led1, OUTPUT);
  pinMode(Led2, OUTPUT);
  pinMode(Led3, OUTPUT);
  pinMode(Led4, OUTPUT);
  pinMode(Led5, OUTPUT);
  pinMode(ButtonPin, INPUT_PULLUP); //Подтягиваем
резистор для кнопки

  Timer1.initialize();//По базе - 1 сек (в микро сек.)

  Serial.begin(9600);
  sei();
}

void Led_Off(){ //Все светодиоды выключены
  for (int i = 8; i<=12;i++) digitalWrite(i, 0);
}

void DelayMode(){ //Гирлянда на delay
  for (int i = 8; i<=12;i++){
    digitalWrite(i, 1);
    delay(100);
    digitalWrite(i, 0);
  }

}

void MillisMode(){ //На millis, два светодиода моргают
поочередно, третий моргает быстро
  if(millis() - led_time > 1000){
    led_time = millis();
    for (int i = 8; i<=12;i++) digitalWrite(i, 0);
  }
}

```

```

    if (Led_pos == 0) digitalWrite(Led1, HIGH);
    else if (Led_pos == 1) digitalWrite(Led2, HIGH);
    else if (Led_pos == 2) digitalWrite(Led3, HIGH);

    Led_pos = (Led_pos + 1) % 3;
}
if(millis() - led_time_3 > 200){
    led_time_3 = millis();
    digitalWrite(Led4, Led_pos_3);
    digitalWrite(Led5, !Led_pos_3);
    Led_pos_3 = !Led_pos_3;
}
}

void TimerMode(){ //Ha Timer
    if (!Led_Zero){
        for (int i = 8; i<=12;i++) digitalWrite(i, 0);
        Led_Zero = true;
        Led_pos_timer = 0;
    }
    if (Led_pos_timer == 0) digitalWrite(Led1, 1);
    else if (Led_pos_timer == 1) digitalWrite(Led4, 1);
    else if (Led_pos_timer == 2) digitalWrite(Led2, 1);
    else if (Led_pos_timer == 3) digitalWrite(Led5, 1);
    else if (Led_pos_timer == 4) digitalWrite(Led3, 1);
    else if (Led_pos_timer == 5) digitalWrite(Led4, 0);
    else if (Led_pos_timer == 6) digitalWrite(Led1, 0);
    else if (Led_pos_timer == 7) digitalWrite(Led5, 0);
    else if (Led_pos_timer == 8) digitalWrite(Led3, 0);
    else if (Led_pos_timer == 9) digitalWrite(Led2, 0);
    Led_pos_timer = (Led_pos_timer + 1) % 10;
}

```

```

void loop() {
    Button = !digitalRead(ButtonPin);
    if(Button == 1 && flag == 0 && millis() - last_time > 100)
    {
        flag = 1;
        Counter++;
        last_time = millis();
    }
    if(Button == 0 && flag == 1) {
        flag = 0;
    }

    if (Counter > 3) Counter = 0;
    switch (Counter) {
        case 0:
            Timer1.detachInterrupt();
            Led_Zero = 0;
            Led_Off();
            break;
        case 1:
            DelayMode();
            break;
        case 2:
            MillisMode();
            break;
        case 3:
            Timer1.attachInterrupt(TimerMode);
            break;
    }
}

```