

Маячок із двома спалахами

LED двічі коротко спалахує, робить довгу паузу й повторює послідовність.

- Arduino Uno, USB-кабель і макетна плата.
- Один червоний світлодіод.
- Один резистор 330 Ом.
- Два з'єднувальні дроти.

Під'єднай LED до **D8** за координатами з наступних кроків.

Для червоного LED можна використати наявний резистор 220 Ом замість 330 Ом; у готовій схемі Wokwi стоїть 330 Ом.

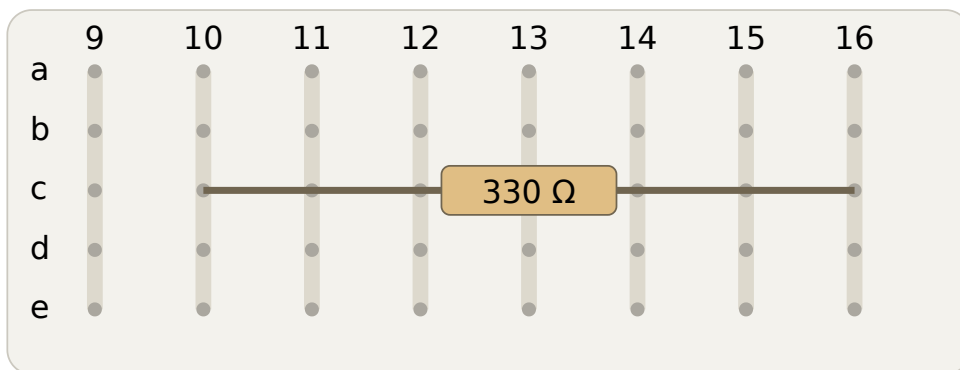
Перед зміною схеми від'єднай живлення. У Wokwi зупини симуляцію. Справжню схему перевір разом з учителем перед підключенням USB.

З'єднання груп 10 та 16

Встав резистор 330 Ом між 10с та 16с.

Цього разу він лежить уздовж однієї половини макетної плати. Групи 10a–10e та 16a–16e окремі, тому струм між ними проходить через резистор.

Обидва кінці в одній групі, наприклад 10a та 10e, не підходять: внутрішній контакт плати обійде резистор.



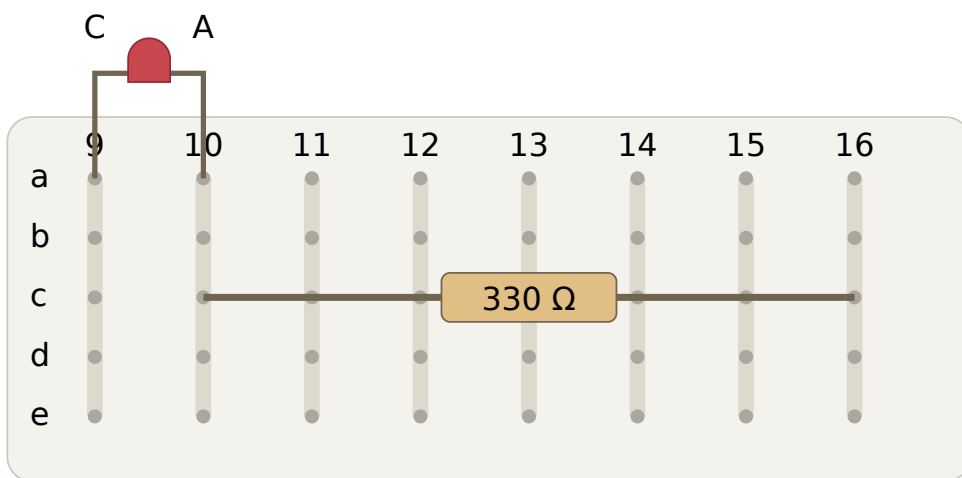
Отвори а–е одного номера з'єднані

Анод і катод

1. Довгу ніжку LED, анод А, встав у **10а**.
2. Коротку ніжку, катод С, встав у **9а**.

Анод з'єднаний із резистором через групу 10. Катод стоїть у сусідній окремій групі 9.

Якщо ніжки обрізані, визнач катод за пласкою гранню корпусу або перевір LED з учителем.



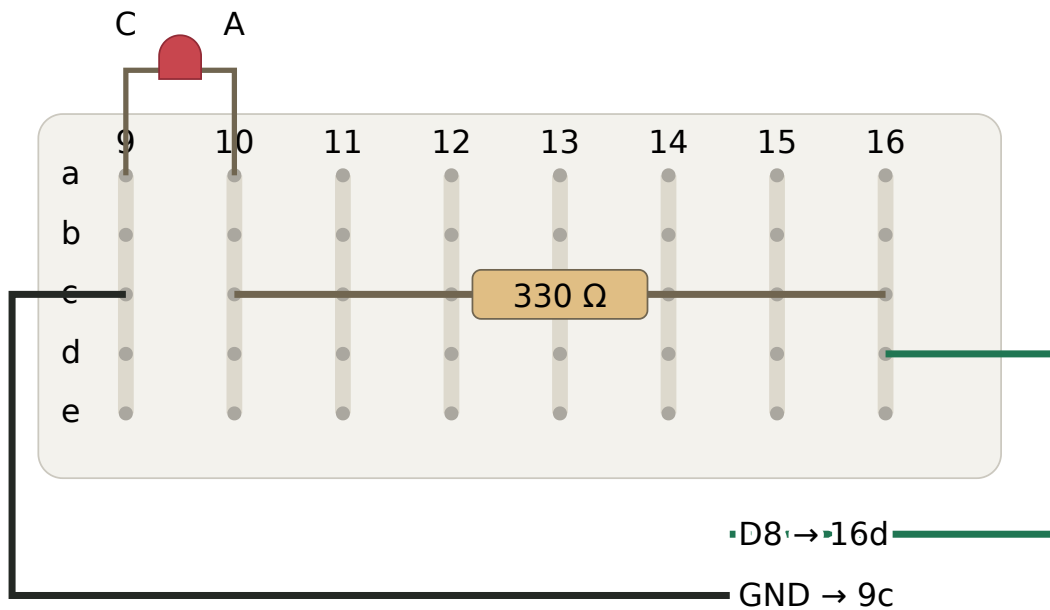
Отвори а-е одного номера з'єднані

D8 та GND

1. З'єднай **D8** Arduino з **16d** зеленим дротом.
2. З'єднай **GND** Arduino з **9c** чорним дротом.

D8 → **16d** → **16c** → **резистор** → **10c** → **10a** → **LED** → **9a** → **9c** → **GND**

Простеж цей шлях на своїй платі. Резистор має бути послідовно зі світлодіодом.



Отвори а-е одного номера з'єднані

Три тривалості

```
const byte LED_PIN = 8; // Пін світлодіода.  
const unsigned long FLASH_MS = 200; // Спалах.  
const unsigned long SHORT_PAUSE_MS = 200; // Коротка пауза.  
const unsigned long LONG_PAUSE_MS = 1200; // Довга пауза.
```

Усі значення часу задані в мілісекундах. 200 мс = 0,2 с; 1200 мс = 1,2 с.
Закінчення `_MS` у назві нагадує одиницю вимірювання.

`const` означає, що значення не змінюється під час виконання. `byte` достатньо для номера піна. `unsigned long` зберігає невід'ємні цілі числа й підходить для тривалостей у мілісекундах.

Описуємо один спалах

```
void flashOnce() { // Дії одного спалаху.  
    digitalWrite(LED_PIN, HIGH); // Увімкнути LED.  
    delay(FLASH_MS); // Почекаати 200 мс.  
    digitalWrite(LED_PIN, LOW); // Вимкнути LED.  
}
```

Функція — група команд із власним ім'ям. `flashOnce()` виконує три дії: увімкнути, зачекати, вимкнути.

Вона **не містить паузи після вимкнення**. Таку паузу задаємо окремо — вона буде короткою після першого спалаху та довгою після другого.

`void` означає, що функція не повертає значення. Її опис розміщуємо поза `setup()` і `loop()`.

Починаємо з вимкненого LED

```
void setup() { // Один раз після запуску.  
  pinMode(LED_PIN, OUTPUT); // Пін 8 — вихід.  
  digitalWrite(LED_PIN, LOW); // LED вимкнений.  
}
```

`setup()` готує вихід. Вона виконується один раз після подачі живлення або перезапуску.

Сам опис `flashOnce()` не запускає спалах. Потрібно викликати функцію — написати її ім'я з дужками та крапкою з комою.

Два виклики та дві паузи

```
void loop() { // Повторюємо всю послідовність.  
  flashOnce(); // Світло 200 мс, потім вимкнення.  
  delay(SHORT_PAUSE_MS); // Темрява 200 мс.  
  flashOnce(); // Ще 200 мс світла, потім вимкнення.  
  delay(LONG_PAUSE_MS); // Темрява 1200 мс.  
}
```

Один цикл триває приблизно **1800 мс**: $200 + 200 + 200 + 1200$. Після останньої паузи виконання повертається до першого спалаху.

Обидва спалахи використовують ту саму функцію. Зміна `FLASH_MS` змінить тривалість обох.

`delay()` зупиняє виконання наступних команд цієї програми на заданий час. Поки триває пауза, світлодіод зберігає свій стан.

Перевір ритм

Запуск у Wokwi

Натисни кнопку запуску. Для самостійного завдання збережи власну копію проєкту.

Готовий проєкт: <https://wokwi.com/projects/475796693291223041>. Схему можна зібрати вручну за кроками уроку. Повний код — на останній сторінці.

Для справжньої плати розпакуй архів коду й відкрий `sketch/sketch.ino` в Arduino IDE, вибери Arduino Uno та порт. Перевір коло з учителем і завантаж код.

1. Знайди два близькі спалахи та довгу темну паузу.
2. Порахуй п'ять повних циклів: це приблизно 9 секунд.
3. Переконайся, що між спалахами LED гасне.

Світлодіод не світиться

Від'єднай живлення. Перевір `D8-16d`, `GND-9c`, резистор `10c-16c` та полярність LED. У коді `LED_PIN` має дорівнювати 8.

Є лише один довгий спалах

Перевір команду `LOW` наприкінці `flashOnce()` та `SHORT_PAUSE_MS`. Без темної паузи два спалахи візуально зливаються.

Подовжи паузу між парами

Зміни довгу паузу з 1200 на 2200 мс. Обидва спалахи й коротку паузу залиш незмінними.

1. Знайди одну константу, яку потрібно змінити.
2. Перед запуском обчисли нову тривалість циклу.
3. Запусти код і порівняй ритм із попереднім.

Пояснення після спроби

`LONG_PAUSE_MS = 2200` . Цикл: $200 + 200 + 200 + 2200 = \mathbf{2800\text{ мс}}$, тобто 2,8 с.
Змінювати `flashOnce()` не потрібно.

Поясни вчителю, чому функцію спалаху викликаємо двічі, а описуємо лише один раз.

Маячок

Два спалахи по 200 мс, коротка пауза 200 мс і довга пауза 1200 мс.

```
#include <Arduino.h> // Команди керування Arduino.

const byte LED_PIN = 8; // Пін світлодіода.
const unsigned long FLASH_MS = 200; // Тривалість одного спалаху в мілісекундах.
const unsigned long SHORT_PAUSE_MS = 200; // Темна пауза між двома спалахами.
const unsigned long LONG_PAUSE_MS = 1200; // Темна пауза між парами спалахів.

void flashOnce() { // Однакові дії для кожного спалаху.
    digitalWrite(LED_PIN, HIGH); // Вмикаємо світлодіод.
    delay(FLASH_MS); // Тримаємо світло протягом 200 мс.
    digitalWrite(LED_PIN, LOW); // Вимикаємо світлодіод.
}

void setup() { // Виконується один раз після запуску.
    pinMode(LED_PIN, OUTPUT); // Налаштовуємо пін 8 як вихід.
    digitalWrite(LED_PIN, LOW); // Починаємо з вимкненого світлодіода.
}

void loop() { // Повторюємо пару спалахів та довгу паузу.
    flashOnce(); // Перший спалах.
    delay(SHORT_PAUSE_MS); // Коротка пауза без світла.
    flashOnce(); // Другий спалах.
    delay(LONG_PAUSE_MS); // Довга пауза перед наступною парою.
}
```