

<https://doi.org/10.31891/2219-9365-2023-74-19>

УДК 004.94

КУЧЕРУК Володимир

Уманський національний університет садівництва  
Вінницький національний аграрний університет  
<https://orcid.org/0000-0002-6422-7779>  
e-mail: [vladimir.kucheruk@gmail.com](mailto:vladimir.kucheruk@gmail.com)

КУЛАКОВ Павло

Уманський національний університет садівництва  
<https://orcid.org/0000-0002-0167-2218>  
e-mail: [kulakovpi@gmail.com](mailto:kulakovpi@gmail.com)

ЛІМАН Віталій

Вінницький національний аграрний університет  
<https://orcid.org/0000-0003-1280-237X>  
e-mail: [limanv@ukr.net](mailto:limanv@ukr.net)

## ОНЛАЙН-СЕРВІС WOKWI ЯК СИМУЛЯТОР МІКРОПРОЦЕСОРНИХ СИСТЕМ

*Мікропроцесорні системи здатні обробляти велику кількість даних і виконувати складні операції з високою точністю і швидкістю, і тому вони використовуються з різноманітною метою, у тому числі для систем Індустрії 4.0 та Індустрії 5.0. Використання мікропроцесорних систем стає все більш актуальним для забезпечення ефективності та якості продукції. Водночас мікропроцесорні системи – це складні пристрої, які потребують відповідного рівня знань та кваліфікації для їх налагодження та обслуговування. Тому їх використання має супроводжуватися підготовкою та підтримкою фахівців. Для цього також можуть бути використані онлайн-сервіси для симуляції мікропроцесорних систем, які дозволяють тестувати та відлагоджувати свої проекти без необхідності мати фізичний пристрій.*

*У даній статті розглядається як приклад моделювання мікропроцесорної системи на основі онлайн-сервісу Wokwi та мікроконтролера ESP32.*

*Ключові слова: Мікропроцесорна система, мікропроцесор, мікроконтролер, програмне забезпечення, онлайн-сервіс.*

KUCHERUK Volodymyr

Uman National University of Horticulture  
Vinnytsia National Agrarian University

KULAKOV Pavlo, LIMAN Vitaly

Vinnytsia National Agrarian University

## WOKWI ONLINE SERVICE AS A SIMULATOR OF MICROPROCESSOR SYSTEMS

*Microprocessor systems (MPS) are electronic devices that are used to process data and control various processes in various fields of activity. They play an important role in many aspects of our lives, from home appliances to complex automation and production control systems.*

*The relevance of using MPS lies in their ability to process a large amount of data and perform complex operations with high accuracy and speed. They can be used to control various parameters such as temperature, humidity, pressure, speed, and much more. With the help of MPS, many processes can be automated, which ensures the efficiency and quality of products, as well as reducing production costs. MPSs are also widely used in medicine, where they are used to monitor the condition of patients, control medications, diagnose diseases, and much more. In addition, MPS are an integral part of the development of the Internet of Things (IoT), where they are used to collect and transmit data from various sensors and devices, which allows you to create a "smart" home or city.*

*Microprocessor systems are able to process large amounts of data and perform complex operations with high accuracy and speed, and therefore they are used for a variety of purposes, including for Industry 4.0 and Industry 5.0 systems. The use of microprocessor systems is becoming more and more relevant to ensure the efficiency and quality of products. At the same time, microprocessor systems are complex devices that require the appropriate level of knowledge and qualifications for their debugging and maintenance. Therefore, their use should be accompanied by the training and support of specialists. Online services for simulating microprocessor systems can also be used for this, allowing you to test and debug your designs without the need to have a physical device.*

*This article considers as an example the modeling of a microprocessor system based on the Wokwi online service and the ESP32 microcontroller.*

*Keywords: Microprocessor system, microprocessor, microcontroller, software, online service.*

### Постановка проблеми у загальному вигляді

#### та її зв'язок із важливими науковими чи практичними завданнями

Мікропроцесорні системи (МПС) - це електронні пристрої, які використовуються для обробки даних і управління різними процесами в різних галузях діяльності. Вони відіграють важливу роль у багатьох аспектах нашого життя, від домашньої техніки до складних систем автоматизації та управління виробництвом.

Актуальність використання МПС полягає в їх здатності обробляти велику кількість даних і виконувати складні операції з високою точністю і швидкістю. Вони можуть бути використані для контролю за різними параметрами, такими як температура, вологість, тиск, швидкість, і багато іншого. За допомогою МПС можна автоматизувати багато процесів, що забезпечує ефективність та якість продукції, а також зниження витрат на виробництво. МПС також знайшли широке застосування в медицині, де вони використовуються для моніторингу стану хворих, контролю за лікарськими препаратами, діагностики захворювань та багато іншого. Крім того, МПС є невід'ємною частиною розвитку інтернету речей (IoT), де вони використовуються для збору і передачі даних з різних датчиків та пристроїв, що дозволяє створити "розумний" будинок або місто.

У світі, де технології розвиваються все швидше, використання МПС стає все більш актуальним. Ці системи є ключовим елементом у багатьох галузях діяльності, від промисловості до медицини та побутової техніки. Вони дозволяють забезпечувати ефективність та якість продукції, розширювати можливості управління та контролю, а також забезпечувати більш точний та швидкий збір та обробку даних.

З урахуванням швидкого розвитку технологій та зростання потреб в автоматизації та управлінні, використання МПС стає все більш необхідним і перспективним. Однак, використання МПС повинно супроводжуватися відповідною підготовкою та дотриманням відповідних стандартів безпеки, щоб забезпечити ефективність та безпеку їх застосування.

Крім того, МПС - це складні технологічні пристрої, які потребують відповідного рівня знань та кваліфікації для їх налагодження та обслуговування. Це означає, що використання МПС повинно супроводжуватися відповідною підготовкою та підтримкою фахівців. На сьогоднішній день існує багато онлайн-сервісів для симуляції мікропроцесорних систем, які дозволяють розробникам, інженерам та студентам тестувати та відлагоджувати свої проекти без необхідності мати фізичний пристрій. Можна навести наступні приклади:

- Wokwi [1] - це безкоштовний онлайн-сервіс для симуляції мікроконтролерних пристроїв. Wokwi має дуже зручний інтерфейс, дозволяє користувачам створювати складні пристрої, та має широкий набір компонентів.

- CircuitVerse [2] - це онлайн-сервіс для розробки та симуляції електронних схем. CircuitVerse має велику бібліотеку компонентів, що дозволяє користувачам створювати складні схеми, та підтримує відлагодження коду.

- Proteus Design Suite [3] - це платформа для проектування та симуляції електронних пристроїв. Proteus Design Suite має розширену бібліотеку компонентів, що дозволяє користувачам створювати складні пристрої, та підтримує відлагодження коду.

- Tinkercad [4] - це онлайн-сервіс для розробки та симуляції електронних схем та пристроїв. Tinkercad має простий інтерфейс та підтримує відлагодження коду.

- Virtual Breadboard [5] - це платформа для проектування та симуляції електронних пристроїв. Virtual Breadboard має широкий набір компонентів, що дозволяє користувачам створювати складні пристрої, та підтримує відлагодження коду.

Ці сервіси дозволяють користувачам тестувати свої проекти без необхідності мати фізичний пристрій, що робить їх особливо цінними для студентів та початківців в галузі електроніки та програмування.

Дослідження популярності цих платформ за допомогою Google Trends показали, що найбільшої популярності у світі набули Wokwi та Tinkercad. Проведемо аналіз їхніх основних характеристик.

Wokwi та Tinkercad - це два онлайн сервіси, які надають можливість симуляції мікропроцесорних систем. Хоча обидва сервіси дозволяють користувачам розробляти та тестувати мікропроцесорні пристрої в віртуальному середовищі, вони мають деякі відмінності у своїх можливостях та функціональності.

Однією з основних відмінностей між Wokwi та Tinkercad є те, що Wokwi спеціалізується на мікропроцесорних системах, тоді як Tinkercad пропонує більш широкий спектр інструментів для розробки електроніки, таких як симуляція логічних схем, макетування плат та інших електронних пристроїв. Тому, якщо користувачі шукають сервіс для симуляції конкретно мікроконтролерних пристроїв, то Wokwi може бути кращим вибором.

Ще однією відмінністю є те, що Wokwi має більш гнучкий та зручний інтерфейс користувача, який дозволяє користувачам розробляти та тестувати мікропроцесорні пристрої шляхом додавання компонентів та з'єднань до схеми, в той час як Tinkercad пропонує готові компоненти та більш обмежену кількість налаштувань.

Ще однією важливою відмінністю є те, що Wokwi дозволяє користувачам взаємодіяти зі своїми пристроями за допомогою JavaScript, що дозволяє створювати складні алгоритми та візуальні ефекти. Tinkercad не має такої можливості. Wokwi є безкоштовним та відкритим, що дозволяє користувачам розробляти та тестувати свої мікропроцесорні пристрої без необхідності в оплаті за підписку або платіж за використання. Тим часом, Tinkercad пропонує безкоштовну версію, але обмежує доступ до деяких функцій та можливостей.

Крім того, Wokwi має відкритий API, що дозволяє розробникам створювати свої власні компоненти та інтегрувати їх у віртуальне середовище, тоді як Tinkercad має більш обмежену кількість компонентів та не дозволяє розробникам створювати свої власні.

Отже, якщо користувачам потрібен сервіс для симуляції конкретно мікроконтролерних пристроїв, то Wokwi може бути кращим вибором завдяки своїй гнучкості та зручному інтерфейсу користувача. Якщо ж користувачам потрібно розробляти більш широкий спектр електронних пристроїв, Tinkercad може бути кращим вибором, оскільки він пропонує більш широкий спектр інструментів та компонентів для розробки електроніки.

Загальний аналіз показав, що онлайн-сервіс Wokwi найкраще підходить до розробки та проектування мікропроцесорних систем.

### Виклад основного матеріалу

Розглянемо використання сервісу Wokwi на прикладі створення конкретного проєкту.

Вибір плат з мікропроцесорами для розробки у сервісі Wokwi наступний:

- Arduino Uno, Mega, Nano;
- ATtiny85;
- Сімейство ESP32;
- Raspberry Pi Pico;
- Franzininho.

При цьому можна задати мову програмування із варіантів: C, JavaScript, microPython, Rust.

Створимо проєкт на ESP32, при цьому буде на екрану зображення, представлене на рис. 1.

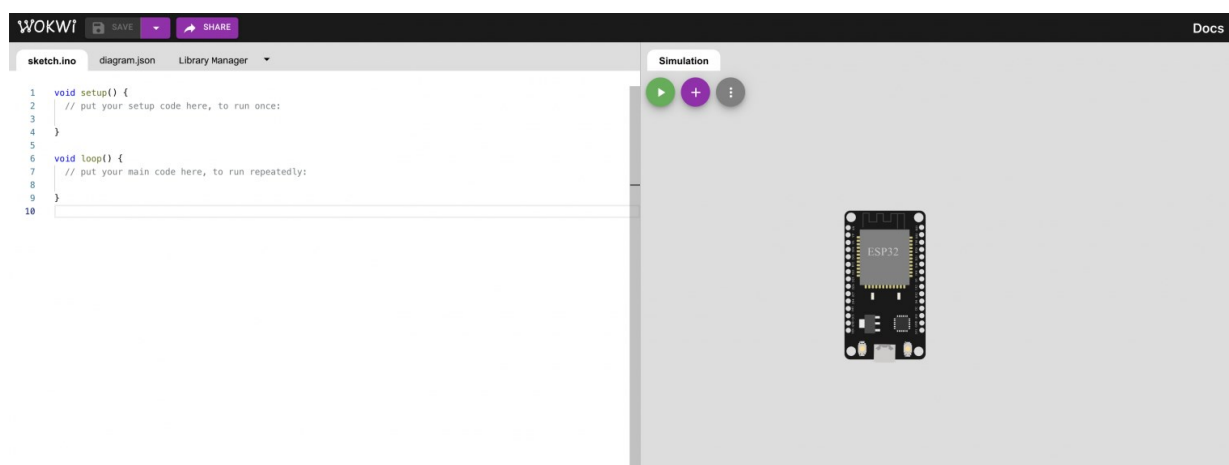


Рис. 1. Початок екрану проєкту на основі EPS32

Зліва – текстовий редактор, в якому можна працювати з файлами програми, а справа – безпосередньо поле симулятора. Наразі на рис. 1 тільки базова плата. Для того, щоб додати елемент, необхідно натиснути на кнопку «+», відкриється список елементів, що випадає.

Створимо простий приклад симуляції – приклад Blink – світлодіод і резистор. Додавимо їх на екран і з’єднаємо. Для того, щоб з’єднати піни і контакти елементів, потрібно натиснути на один контакт, а потім на інший (рис. 2).

Після додавання елементу на поле, якщо навести на нього вказник, появляється рамка з знаком питання. Натиснувши на знак питання, можна потрапити на сторінку документації того чи іншого елементу. Тепер в лівій частині вікна додавимо стандартний приклад Blink, при цьому не забувши вказати потрібний пін. Текст відповідної програми наступний:

```
#define LED_PIN 27

void setup() {
  pinMode(LED_PIN, OUTPUT);
}

void loop() {
  digitalWrite(LED_PIN, HIGH);
}
```

```
delay(1000);  
digitalWrite(LED_PIN, LOW);  
delay(1000);  
}
```

Запустимо симуляцію, натиснувши на зелену кнопку (рис. 3)

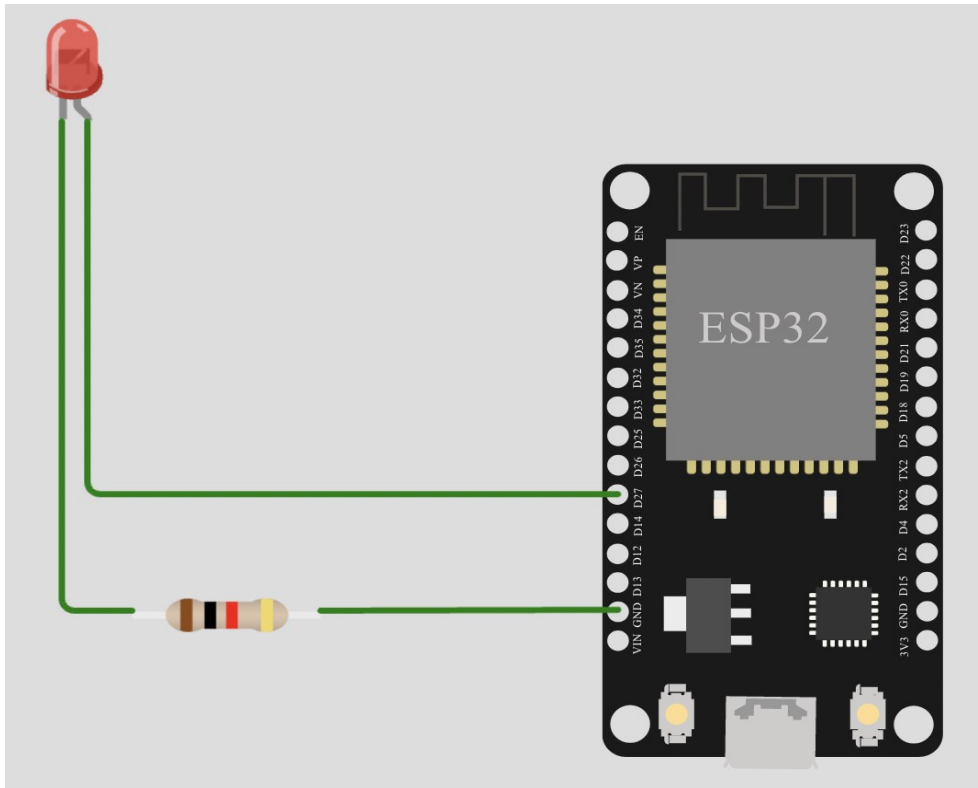


Рис. 2. Створення схеми для симуляції проєкту

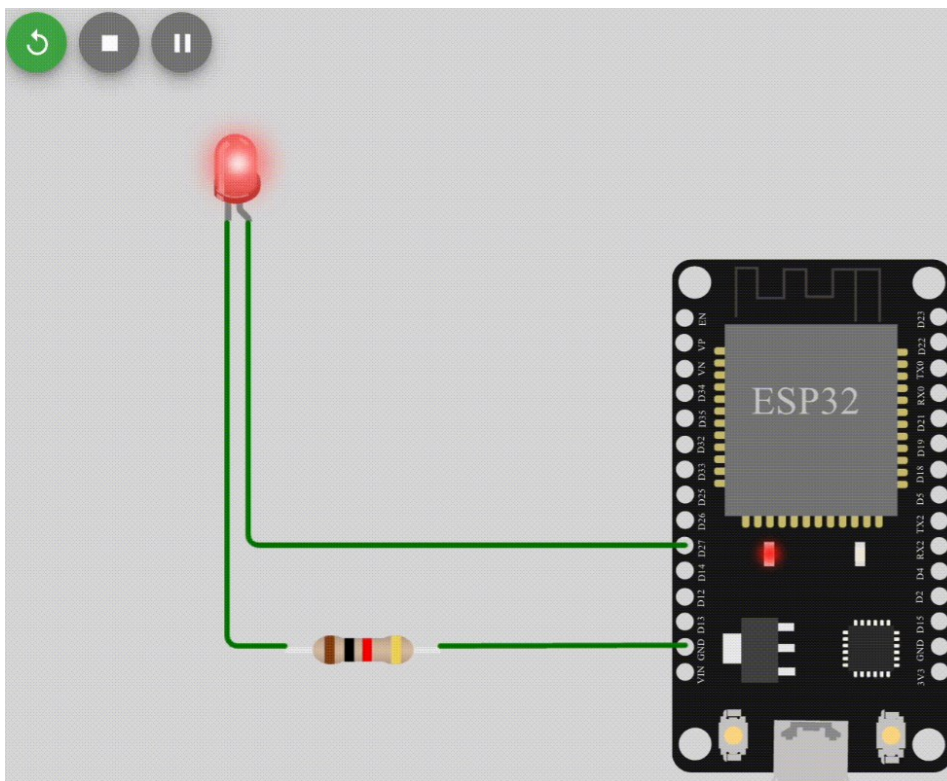


Рис. 3. Запуск симуляції

В процесі проектування можна змінювати налаштування елементів – шляхом редагування файлу `giagram.json` в текстовому редакторі симулятора:

```
{
  "version": 1,
  "author": "Anonymous maker",
  "editor": "wokwi",
  "parts": [
    { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": 0, "left":
0, "attrs": {} },
    {
      "type": "wokwi-led",
      "id": "led1",
      "top": -54.37,
      "left": -185.17,
      "attrs": { "color": "red" }
    },
    { "type": "wokwi-resistor", "id": "r1", "top": 140.29, "left": -
147.16, "attrs": {} }
  ],
  "connections": [
    [ "esp:TX0", "$serialMonitor:RX", "", [] ],
    [ "esp:RX0", "$serialMonitor:TX", "", [] ],
    [ "led1:C", "r1:1", "green", [ "v0" ] ],
    [ "r1:2", "esp:GND.2", "green", [ "v0" ] ],
    [ "led1:A", "esp:D27", "green", [ "v0" ] ]
  ]
}
```

Найчастіше потрібно редагувати об'єкти, містяться в блоці `parts` (опис елементів) і в блоці `connections` (опис з'єднань). Про атрибути, які можна змінювати у світлодіода та резистора, можна дізнатись в [6, 7]. Замінімо колір світлодіода і колір проводів (чорний – земляний, червоний – живлення (пін 27)), тоді файл `giagram.json` прийме вигляд:

```
{
  "version": 1,
  "author": "Anonymous maker",
  "editor": "wokwi",
  "parts": [
    { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": 0, "left":
0, "attrs": {} },
    {
      "type": "wokwi-led",
      "id": "led1",
      "top": -54.37,
      "left": -185.17,
      "attrs": { "color": "blue" }
    },
    { "type": "wokwi-resistor", "id": "r1", "top": 140.29, "left": -
147.16, "attrs": {} }
  ],
  "connections": [
    [ "esp:TX0", "$serialMonitor:RX", "", [] ],
    [ "esp:RX0", "$serialMonitor:TX", "", [] ],
    [ "led1:C", "r1:1", "black", [ "v0" ] ],
    [ "r1:2", "esp:GND.2", "black", [ "v0" ] ],
    [ "led1:A", "esp:D27", "red", [ "v0" ] ]
  ]
}
```

Зображення схеми після заміни атрибутів буде виглядати як на рис. 4.

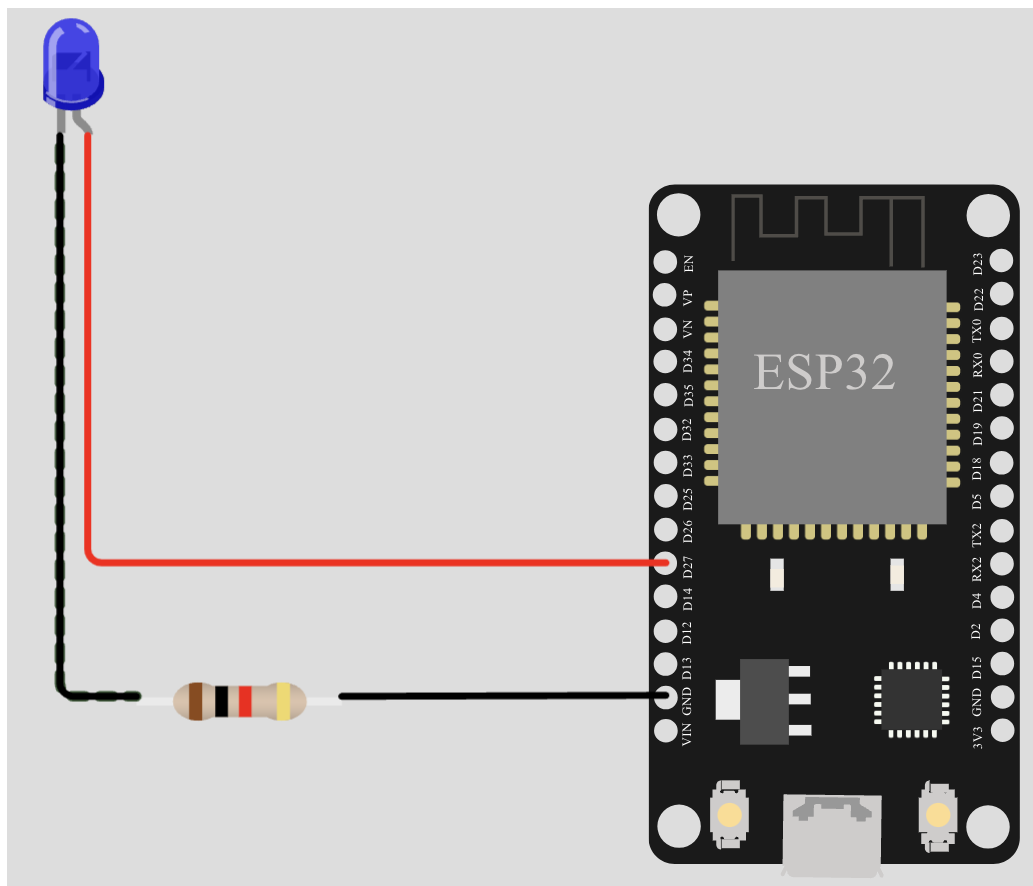


Рис. 4. Зображення схеми після зміни атрибутів елементів

Для користувача доступні велика кількість бібліотек та функцій для роботи з різними периферійними засобами. Для ESP32 також доступна можливість підключення до інтернету. На сайті [1] представлено багато прикладів для роботи з бібліотеками різних пристроїв.

#### Висновки з даного дослідження і перспективи подальших розвідок у даному напрямі

Розглянуто існуючі методи та сервіси онлайн проектування мікропроцесорних систем. Показано на конкретному проекті, що за допомогою таких сервісів можна проектувати апаратне та програмне забезпечення МПС, а також такі сервіси мають перспективу використання в навчальному процесі при умовах відсутності наявного в лабораторіях апаратного забезпечення МПС.

#### Література

1. <https://wokwi.com/>
2. <https://circuitverse.org/>
3. <https://www.labcenter.com/>
4. <https://www.tinkercad.com/>
5. <https://www.virtualbreadboard.com/>
6. <https://docs.wokwi.com/parts/wokwi-led>
7. <https://docs.wokwi.com/parts/wokwi-resistor>

#### References

1. <https://wokwi.com/>
2. <https://circuitverse.org/>
3. <https://www.labcenter.com/>
4. <https://www.tinkercad.com/>
5. <https://www.virtualbreadboard.com/>
6. <https://docs.wokwi.com/parts/wokwi-led>
7. <https://docs.wokwi.com/parts/wokwi-resistor>