

**УКРАЇНСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ  
ЗАЛІЗНИЧНОГО ТРАНСПОРТУ**

**ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КЕРУЮЧИХ СИСТЕМ  
ТА ТЕХНОЛОГІЙ**

**Кафедра автоматики та комп'ютерного телекерування  
рухом поїздів**

**МОДЕЛЮВАННЯ ТА РОБОТА З ДАТЧИКАМИ  
І ВИКОНАВЧИМИ ПРИСТРОЯМИ РОБОТОТЕХНІКИ**

**МЕТОДИЧНІ ВКАЗІВКИ  
до практичних робіт і самостійної роботи**

**з дисципліни  
*«ТЕОРЕТИЧНІ ОСНОВИ РОБОТОТЕХНІКИ»***

**Харків – 2024**

Методичні вказівки розглянуто і рекомендовано до друку на засіданні кафедри автоматики та комп'ютерного телекерування рухом поїздів 25 березня 2024 р., протокол № 8.

Наведено короткі теоретичні відомості, завдання і порядок дій для виконання практичних робіт, а також рекомендації до самостійного вивчення відповідних розділів курсу.

Методичні вказівки призначені для здобувачів вищої освіти другого (магістерського) рівня спеціальності 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка», які вивчають дисципліну «Теоретичні основи робототехніки», усіх форм здобуття освіти.

Укладачі:

доценти С. О. Змій,

О. В. Щебликіна,

І. М. Сіроклин,

старш. викл. М. В. Ушаков

Рецензент

проф. В. І. Мойсеєнко

## ЗМІСТ

ВСТУП.....	4
ПРАКТИЧНА РОБОТА 1. Використання резистивного датчика тиску .....	5
ПРАКТИЧНА РОБОТА 2. Використання датчиків нахилу.....	12
ПРАКТИЧНА РОБОТА 3. Використання датчиків вологості.....	16
ПРАКТИЧНА РОБОТА 4. Використання газових сенсорів (датчиків) .....	22
ПРАКТИЧНА РОБОТА 5. Використання фоторезисторів (датчиків освітленості) .....	28
СПИСОК ЛІТЕРАТУРИ.....	34
Додаток А Основи роботи в Tinkercad.....	35

## ВСТУП

Метою методичних вказівок є розроблення циклу практичних робіт різного формату з використанням сенсорних пристроїв, датчиків-перетворювачів первинної інформації та виконавчих пристроїв, які можуть бути використані як у робототехніці, так і для ознайомлення з сучасними тенденціями розвитку електроніки, вимірювальної та обчислювальної техніки в галузі сенсорних пристроїв; прийомами автоматизації, контролю параметрів середовища. Об'єктом дослідження є датчики, сенсорні та виконавчі пристрої. Для досягнення поставленої мети використовували безкоштовний вебдодаток для 3D-проектування, роботи з електронними компонентами і написання програмного коду tinkercad і програмування пристроїв Arduino, що базується на C++.

# ПРАКТИЧНА РОБОТА 1

## Використання резистивного датчика тиску

### 1.1 Мета роботи

Розроблення програми використання резистивного датчика, що може бути використано в робототехніці, наприклад маніпуляторах.

### 1.2 Короткі теоретичні відомості

Резистивний датчик тиску (сили) (Force Sensitive Resistors, FSRs) – це датчик, опір якого змінюється залежно від сили, прикладеної до чутливого елемента. Цей датчик параметричного типу, за допомогою якого можна оцінити рівень тиску, силу натискання або вагу. На рисунку 1.1 показано один із найпоширеніших видів датчиків, що випускає фірма Sparkfun. Радіус чутливого елемента – 15 мм.



Рисунок 1.1 – Зовнішній вигляд резистивного датчика тиску

Датчик має два шари, розділені ізоляційною обкладкою. Верхній шар складається з плівки із зустрічно надрукованими провідниками, який є чутливим елементом. Нижній шар – підкладка з друкованим напівпровідником. Зі збільшенням сили натискання на датчик збільшується

і площа зіткнення двох шарів, що буде знижувати опір датчика. На рисунку 1.2 подано структуру резистивного датчика тиску (сили).

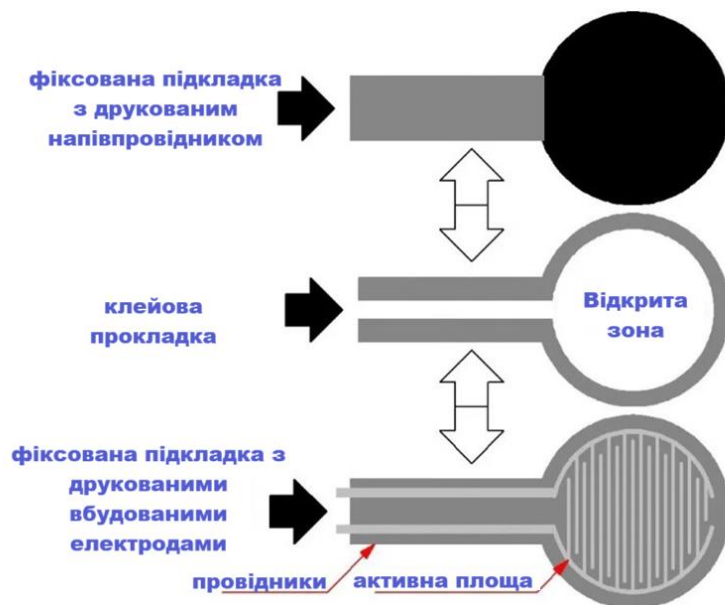


Рисунок 1.2 – Конструкція датчика тиску

Недоліком цього датчика є неточність вимірювань і нелінійність зміни опору. На рисунку 1.3 зображено графік зміни опору залежно від прикладеної сили або тиску.

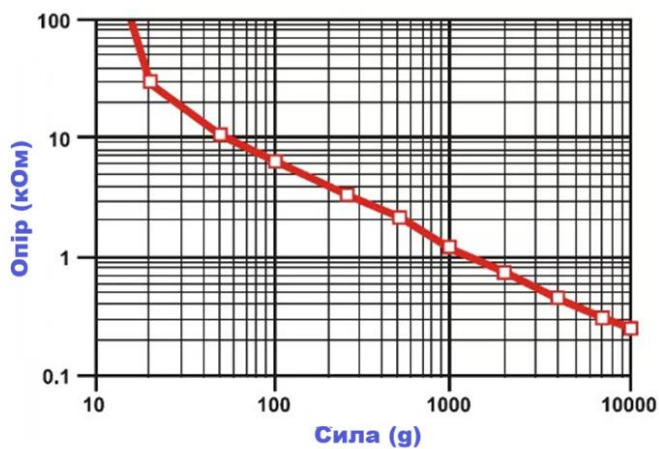


Рисунок 1.3 – Графік зміни опору від прикладеної сили або тиску

На рисунку 1.4 подано схему з'єднання резистивного датчика тиску з резистором, приєднаним як дільник напруги. Зроблено це для контролю вхідної напруги і зняття показань з датчика.

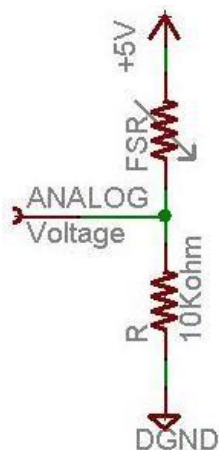


Рисунок 1.4 – Принципова схема підключення резистивного датчика тиску

У таблиці 1.1 наведено приблизні значення аналогової напруги під час роботи резистивного датчика тиску з живленням від 5 В і знизувальним резистором 10 кОм в електричному колі.

Таблиця 1.1 – Приблизні значення аналогової напруги під час роботи резистивного датчика тиску з живленням від 5 В і знизувальним резистором 10 кОм в електричному колі

Сила, Н	Опір $FSR$	Опір $FSR+R$	Сила струму через $FSR+R$	Напруга через $R$
Немає	$\infty$	$\infty$	0 мА	0 В
0,2 Н	30 кОм	40 кОм	0,13 мА	1,3 В
1 Н	6 кОм	16 кОм	0,31 мА	3,1 В
10 Н	1 кОм	11 кОм	0,45 мА	4,5 В
100 Н	250 Ом	10,25 кОм	0,49 мА	4,9 В

Рівняння розрахунку напруги у схемі з резистором, підключеним як дільник напруги, має вигляд

$$V_o = V_{cc} \cdot \frac{R}{R+FSR}.$$

Технічні характеристики датчика:

- діапазон фіксування ваги: 100 г...10 кг;
- довжина: 61 мм;
- діаметр підкладки з чутливим елементом: 18 мм; товщина: 0,2 мм;
- вага: 20 г;
- гарантований опір без чинення тиску: > 1 МОм;
- діапазон опорів: ∞...200 Ом;
- зносостійкість: до 10 мільйонів натискань;
- діапазон робочих температур: -30 °С...+70 °С;
- максимальна швидкість спрацьовування: 1-2 мс;
- максимальний струм: I<sub>ma</sub> / см<sup>2</sup> прикладеної сили;
- відстань між контактами: 2,54 мм.

Можливості програми наведено в додатку А.

### 1.3 Порядок виконання

Для виконання практичного завдання потрібно зайти на сайт [www.tinkercad.com](http://www.tinkercad.com), потім вибрати створення кола. Можливості програми (сайту) наведено в додатку А. Обрати такі компоненти: arduino uno, світлодіоди – 5 штук, резистори – 6 штук, п'єзоелемент, резистивний датчик тиску та макетну плату.

Підключити такі елементи:

а) аноди п'яти світлодіодів через обмежувальні резистори до п'яти цифрових пінів arduino, а катоди до землі (GND) arduino;



б) п'єзоелемент. Плюс (+) підключити до цифрового піна arduino, а мінус – до землі (GND) arduino;

в) резистивний датчик тиску не має полярності, тому неважливо, який контакт буде прийнятий за плюс (+), а який – за мінус (-). Контакт, прийнятий за плюс (+), під'єднати до живлення 5 В arduino, а контакт, прийнятий за мінус, через знижувальний резистор 10 кОм під'єднати до землі (він виступатиме як дільник напруги). Потім точку кола між резистором і резистивним датчиком тиску (змінним резистором) під'єднати до аналогового піна для зняття показань із датчика.

Написати скетч, що реалізує такі дії: у разі збільшення тиску на датчик спалахують послідовно світлодіоди, починаючи з третього світлодіода ввімкнути п'єзоелемент і залишити його ввімкненим до останнього світлодіода, п'ятого. Також цю схему можна зібрати в очному форматі, використовуючи arduino uno і кабель для завантаження скетчу.

На рядках 3-9 задаємо режими роботи пінів, до яких підключені елементи. Значення резистивного датчика тиску зчитуємо з аналогового піна A0 (рядок 13).

Лістинг програми (скетч):

```
1. int h = 0;
2. void setup () {
3.   pinMode(A0, INPUT);
4.   pinMode(13, OUTPUT);
5.   pinMode(12, OUTPUT);
6.   pinMode(11, OUTPUT);
7.   pinMode(10, OUTPUT);
8.   pinMode(9, OUTPUT);
9.   pinMode(8, OUTPUT);
10. Serial.begin (9600);
11. }
12. void loop() {
13.   h = analogRead(A0);
14.   if(h >100) {
15.     digitalWrite(13, HIGH);
16.     digitalWrite (8, LOW) ;
17.   }else{
```

```

18. digitalWrite(13, LOW);
19. digitalWrite(8, LOW);
20. }
21. if(h >200) {
22. digitalWrite(12, HIGH);
23. digitalWrite(8, LOW);
24. }else{
25. digitalWrite (12, LOW);
26. digitalWrite(8, LOW);
27. }
28. if(h >300) {
29. digitalWrite(11, HIGH);
30. digitalWrite(8, HIGH);
31. }else{
32. digitalWrite(11, LOW);
33. digitalWrite(8, LOW);
34. }
35. if(h >400) {
36. digitalWrite(10, HIGH);
37. digitalWrite(8, HIGH);
38. }else{
39. digitalWrite(10, LOW);
40. digitalWrite(8, LOW);
41. }
42. if(h >450) {
43. digitalWrite(9, HIGH);
44. digitalWrite(8, HIGH);
45. }else{
46. digitalWrite(9, LOW);
47. digitalWrite(8, LOW);
48. }
49. Serial.println(h);
50. delay(1000);
51. }

```

За значень вище 100 подається високий сигнал на тринадцятий пін (HIGH), на другий пін подається низький сигнал (LOW), інакше (else) подається на восьмий і тринадцятий піни – низький сигнал (LOW) (14-20 рядки). За значень вище 200 подається високий сигнал на тринадцятий пін (HIGH), на восьмий – низький сигнал (LOW), інакше (else) подається на восьмий і тринадцятий піни – низький сигнал (LOW) (21-27 рядки). За

значеннях вище 300 подається високий сигнал на восьмий і одинадцятий піни (HIGH), інакше (else) подається на восьмий і одинадцятий піни низький сигнал (LOW) (28-34 рядки). За значеннях вище 400 подається високий сигнал (HIGH) на восьмий і десятий піни, інакше (else) подається на восьмий і десятий піни низький сигнал (LOW) (35-41 рядки). За значень вище 450 подається високий сигнал (HIGH) на восьмий і дев'яти піни, інакше (else) подається на восьмий і дев'ятий піни низький сигнал (LOW) (42-47 рядки). Виведення значень "h" на монітор порту й очікування 1000 мс (рядки 49 та 50).

Запустити моделювання та перевірити роботу датчика.

### **Запитання для самоперевірки**

1 Як змінюється опір резистивного датчика тиску при збільшенні тиску на нього?

2 Як змінюється площа дотику між обкладками резистивного датчика тиску в разі зменшення його опору?

3 Чи має резистивний датчик тиску полярність?

4 Яке значення аналогової напруги буде під час роботи резистивного датчика тиску з живленням від 5 В і знижувальним резистором 10 кОм в електричному колі, якщо докласти силу в 10 Н?

5 Якого типу резистивний датчик тиску?

## ПРАКТИЧНА РОБОТА 2

### Використання датчиків нахилу

#### 2.1 Мета роботи

Розроблення програми використання датчика нахилу, що може бути використано в робототехніці, наприклад маніпуляторах.

#### 2.2 Короткі теоретичні відомості

Датчик нахилу дає змогу визначити відхилення відносно вертикальної площини. Принцип роботи датчика нахилу показано на рисунку 2.1. Цей датчик параметричного типу.

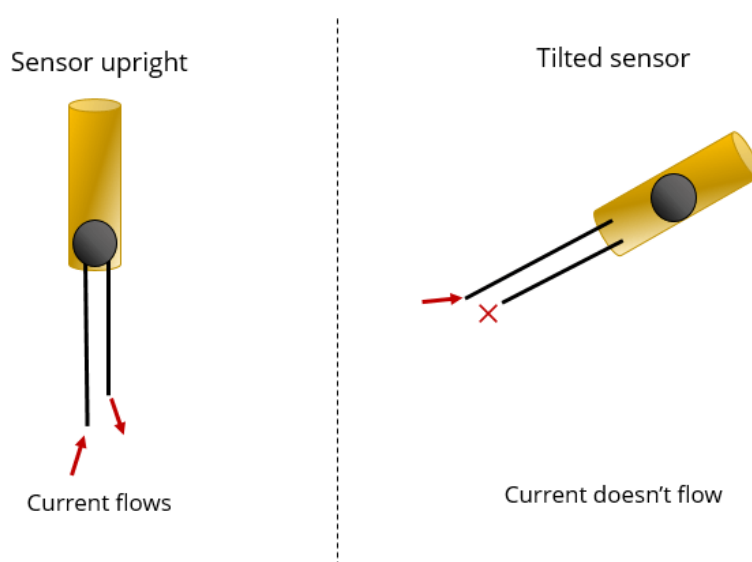


Рисунок 2.1 – Принцип роботи датчика нахилу

Датчик нахилу містить циліндр із струмопровідною кулькою, що перекочується всередині, і двома контактами, що проводять знизу. Залежно від нахилу датчика кулька або замикає контакти, або ні, виступаючи в ролі певного перемикача. Отже, під час під'єднання до Arduino цей датчик видаватиме цифровий сигнал: низький і високий рівні сигналу (0 і 1).

Найпоширеніша модель – це SW-200D (рисунок 2.2). У датчика один вивід сріблястий, а другий – золотистий.

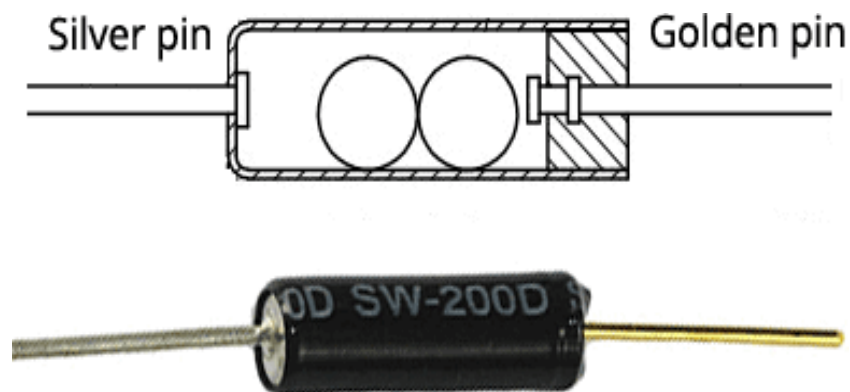


Рисунок 2.2 – Схема роботи датчика нахилу

Технічні характеристики:

- час відгуку: 2 мс;
- опір при замкненому колі: до 10 Ом;
- опір при розімкненому колі: понад 10 мОм;
- робоча напруга: до 12 В;
- споживаний струм: до 20 мА;
- допустима температура: від -40 °С до +80 °С;
- допустима вологість: 95 % RH (40 °С до 96 год);
- зусилля відриву: 500 гс на 1 хв;
- час життя: 100000 циклів; розмір: 12 x 3,6 мм;
- вага: менше 1 г.

### 2.3 Порядок виконання

Для виконання практичного завдання потрібно зайти на сайт [www.tinkercad.com](http://www.tinkercad.com), потім вибрати створення кола. Обрати такі компоненти: arduino uno, світлодіод, резистори – 2 шт., п'єзоелемент, датчик нахилу, мікросервопривод і макетну плату.

Підключити такі елементи:

а) анод світлодіода через обмежувальний резистор до цифрового піна arduino, а катод до землі (GND) arduino;

б) п'єзоелемент. Мінус (-) до землі (GND) arduino, а плюс (+) до цифрового піна arduino;

в) мікросервопривод. Сигнал (S) під'єднати до цифрового піна arduino, плюс (+) – до живлення 5 В arduino, а мінус (-) – до землі (GND) arduino;

г) датчик нахилу. Один контакт під'єднати до живлення 5 В arduino (червоний дріт), а інший контакт через знижувальний резистор до цифрового піна arduino (зелений дріт), інший кінець резистора 10 кОм під'єднати до землі (GND) arduino (чорний дріт). На рисунку 2.3 подано схему підключення датчика нахилу.

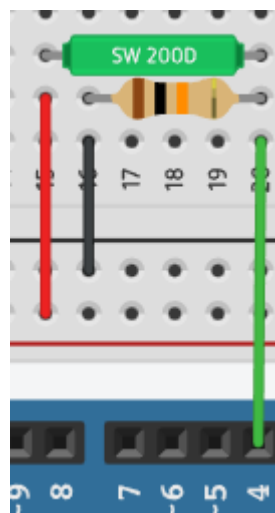


Рисунок 2.3 – Схема підключення датчика нахилу

Написати код програми, що реалізує такі дії: у разі зміни нахилу датчика нахилу спалахує світлодіод, вмикається п'єзоелемент і повертається мікросервопривод на 180°. Також можна зібрати цю схему в очному форматі, знадобиться arduino і кабель для завантаження скетчу. Як альтернативу буде запропоновано датчик вібрації, конструкція якого схожа

з датчиком нахилу, цей датчик має три контакти: плюс (+), мінус (-), сигнал (S). Підключення: сигнал (S) до живлення 5 В arduino, а мінус (-) через знижувальний резистор 10 кОм, як і у схемі з датчиком нахилу, контакт живлення (+) нікуди не підключаємо.

Скетч для роботи з датчиком нахилу:

```
1. #include <Servo.h>
2. Servo myservo;
3. const int ledPin = 12;
4. const int piezo =2;
5. void setup()
6. {
7.     myservo.attach(3);
8.     pinMode(ledPin, OUTPUT);
9.     pinMode(piezo, OUTPUT);
10.    pinMode(4,INPUT);
11.    Serial.begin(9600);
12.    myservo.write(0);
13. }
14. void loop() {
15.     if(digitalRead(4)){
16.         myservo.write(180);
17.     }
18.     else{
19.         myservo.write (0);
20.         if (digitalRead (4)-LOW) {
21.             digitalWrite(12, LOW);
22.             digitalWrite(2, LOW);
23.         } else {
24.             digitalWrite(12, HIGH);
25.             digitalWrite(2, HIGH);
26.         }
27.     }
28. }
```

На рядку 1 додається бібліотека для керування сервоприводами. На рядках 3, 4 вказуємо піни, до яких підключені світлодіод і п'єзоелемент. За допомогою attach() вказуємо пін, до якого під'єднано мікросервопривод (рядок 7). На рядках 8-10 вказуємо режими роботи пінів, до яких під'єднані елементи. На рядку 12 виставляємо мікросервопривод на 0° за допомогою

write(), яка повертає сервопривод на заданий кут. Якщо на цифровий пін подано високий сигнал (HIGH), до якого під'єднано датчик нахилу, то мікросервопривод повертається на 180°, інакше (else) виставляємо в 0. Також якщо на цифровому піні нема високого сигналу, то світлодіод не горить, і п'єзоелемент не ввімкнено, інакше (else) світлодіод спалахує, і вмикається п'єзоелемент (рядки 24, 25).

Запустити моделювання та перевірити роботу датчика.

### **Запитання для самоконтролю**

- 1 Якого типу є датчик нахилу?
- 2 Яка функція attach() для сервоприводів?
- 3 Яка функція write() для сервоприводів?
- 4 На рядках 8, 9 і 10 який режим роботи для пінів задано?
- 5 Який максимальний кут повороту мікросервопривода в tinkercad?

## **ПРАКТИЧНА РОБОТА 3**

### **Використання датчиків вологості**

#### **3.1 Мета роботи**

Розроблення програми використання датчика вологості, що може бути використано в робототехніці.

#### **3.2 Короткі теоретичні відомості**

Датчик вологості ґрунту дає змогу вимірювати рівень вологості ґрунту. Цей датчик параметричного типу. Найпоширенішими датчиками вологості ґрунту є ємнісні та кондуктометричні (резистивні). Ємнісні датчики вологості ґрунту вважаються більш точними і не окиснюються з



часом, як резистивні. Однак у системі Tinkercad є тільки резистивний датчик вологості ґрунту.

Датчик вологості ґрунту складається з двох елементів: зонда (рисунок 3.1) і модуля. Датчик містить вилючний зонд із двома провідниками, який занурюють у ділянку, у якій проводять вимірювання. Провідники в ролі змінного резистора (потенціометр) змінюють опір залежно від вологості ґрунту. Вологий ґрунт покращує провідність, що призводить до зменшення опору, а сухий – до збільшення.



Рисунок 3.1 – Зонд датчика вологості ґрунту

Модуль, який з'єднує датчик з arduino, видає вихідну напругу залежно від зміни опору, яка доступна на виводі аналогового піна (A0). Також сигнал подається на високоточний компаратор LM393 для його оцифрування, з виходу якого сигнал подається на вивід цифрового виходу (DO) (рисунок 3.2). Вбудований потенціометр дає змогу регулювати чутливість цифрового виходу (D0), за допомогою якого можна виставляти порогові значення, перевищивши яке, модуль видає низький логічний рівень, а в інших випадках подається високий. Компаратор містить два світлодіоди, один із яких відповідає за наявність поданої напруги (індикатор живлення), а другий (індикатор стану) – за перевищення порогового значення



Рисунок 3.2 – Регулювання чутливості датчика вологості ґрунту

### 3.3 Порядок виконання

Для виконання практичного завдання необхідно виконати низку дій:

- зайти на сайт [www.tinkercad.com](http://www.tinkercad.com);
- вибрати елемент створення кола;
- вибрати такі компоненти: arduino uno, світлодіоди – 5 шт., резистори – 5 шт., п'єзoeлемент, датчик вологості ґрунту, мікросервопривод і макетну плату.

Підключити такі елементи:

а) аноди п'яти світлодіодів через обмежувальний резистор на п'ять цифрових пінів arduino, а катоди – до землі (GND) arduino;

б) п'єзoeлемент. Мінус (-) підключити до землі (GND) arduino, а плюс (+) до цифрового піна arduino;

в) мікросервопривод. Сигнал (S) під'єднати до цифрового піна arduino, плюс (+) – до живлення 5 В arduino, а мінус (-) – до землі (GND) arduino;

г) датчик вологості ґрунту. Живлення під'єднати до живлення 5 В arduino, але можна під'єднати і до аналогового піна arduino, оскільки за допомогою скетчу з аналогового піна arduino буде подано напругу на датчик, землю – до землі (GND) arduino, а сигнал – до аналогового піна arduino.

Написати код програми, що реалізує такі дії, поділений на кілька етапів: на початковому етапі, коли ґрунт сухий, блимає світлодіод 1, мікросервопривод повернутий на 180°. Зі збільшенням вологості ґрунту світлодіод 1 перестає блимати і загоряється, п'єзоелемент вмикається, мікросервопривод виставляється у вихідне положення (0°). Потім загоряється другий світлодіод, гасне перший, п'єзоелемент вимикається. Після цього загоряється світлодіод 3, гасне світлодіод 2, і так до останнього.

```
#include <Servo.h>
Servo myservo;
int a = 0;
void setup()
{
    myservo.attach (6) ;
    pinMode (A0, OUTPUT);
    pinMode(6, OUTPUT);
    pinMode(7, OUTPUT);
    pinMode(8, OUTPUT);
    pinMode(9, OUTPUT);
    pinMode(10, OUTPUT);
    pinMode(11, OUTPUT);
    pinMode(12, OUTPUT);
}
void loop()
{
    digitalWrite(A0, HIGH);
    delay(10);
    m = analogRead(A1);
    digitalWrite{A0, LOW};
    digitalWrite{6, LOW} ;
    digitalWrite(7, LOW);
    digitalWrite{8, LOW};
    digitalWrite{9, LOW};
    digitalWrite(10, LOW);
    digitalWrite(11, LOW);
    digitalWrite(12, LOW);
    if (m < 100) {
        myservo.write(ISO);
        digitalWrite(12, HIGH);
        delay(100);
        digitalWrite(12, LOW);
    }
}
```

```

    } else {
        myservo.write (0) ;
        if (m < 200) {
            digitalWrite(12, HIGH);
            digitalWrite(7, HIGH);
        } else {
            if (m < 400) {
                digitalWrite(11, HIGH);
            } else {
                if (m < 600) {
                    digitalWrite(10, HIGH);
                } else {
                    if (m < 800) {
                        digitalWrite(S, HIGH);
                    } else {
                        digitalWrite(S, HIGH);
                    }
                }
            }
        }
    }
}
delay(100);
}

```

На рядку 1 вмикаємо бібліотеку для роботи з сервоприводами. На рядках 6-14 вибираємо режими роботи пінів, до яких під'єднано елементи. Подається напруга з аналогового піна датчик (рядок 18). Значення з датчика знімають з аналогового піна (рядок 20). Подача низького сигналу на наявні піни (рядки 21-28).

Якщо значення менше 100, то блимає світлодіод 1, мікросервопривод повернутий на 180° (рядки 29-33), інакше (else) мікросервопривод повернутий у вихідне положення (0°).

Якщо значення менше 200, то загоряється світлодіод 1, вмикається п'єзоелемент (рядки 36-38), інакше (else), якщо значення менше 400, то загоряється світлодіод 2 (рядки 39-41), інакше (else), якщо значення менше 600, то загоряється світлодіод 3 (рядки 42-44), інакше (else), якщо значення

менше 800, то загоряється світлодіод 4 (рядки 45-47), інакше (else) загоряється світлодіод 5.

Також можна зібрати цю схему в очному форматі. Для зручності можна використовувати склянку з водою. Компаратор має чотири контакти: живлення (Vcc), земля (GND), аналоговий контакт (A0) і цифровий контакт (D0), який підключають до відповідних контактів arduino. Реальні значення датчика відрізняються від значень tinkercad, тому для початку важливо дізнатися їх. Підключаємо датчик вологості ґрунту до arduino: цифровий контакт (D0) – до цифрового піна, землю (GND) – до землі (GND) arduino, а живлення (Vcc) – до живлення 5 В arduino.

```
#define sensorPower 7
#define sensorPin A0
void setup ()
{
    pinMode(sensorPower, OUTPUT);
    digitalWrite(sensorPower, LOW);
    Serial.begin(9600);
}
void loop()
{
    Serial.print ("Analog output: ")
    Serial.println(readSensor());
    delay(1000);
}
int readSensor ()
{
    digitalWrite(sensorPower, HIGH)
    delay(10);
    int val = analogRead(sensorPin)
    digitalWrite(sensorPower, LOW);
    return val;
}
```

На рядках 1, 2 вказано виводи, підключені до датчика. Задаємо режим роботи піна для датчика, і датчик залишається увімкненим (рядки 5, 6). На рядках 11, 12 отримуємо показання з функції і друкуємо його. Функція на

рядку 15 повертає аналоговий результат вимірювань датчика вологості ґрунту. На рядку 17 датчик вмикається, читаємо аналогове значення від датчика (рядок 19), потім датчик вимикається, і аналогове значення вологості повертається (рядки 20, 21).

Дізнавшись реальні значення датчика, і якщо високі показання вказують на високий опір (сухий ґрунт), то в первісному скетчі міняємо знаки менше на знаки більше (рядки 29, 36, 40, 43, 46), вставляємо реальні значення і перевіряємо працездатність експериментальної схеми.

Запустити моделювання та перевірити роботу датчика.

### **Запитання для самоконтролю**

- 1 Який тип датчика вологості ґрунту використовують в tinkercad?
- 2 Як змінюється опір датчика у вологому ґрунті?
- 3 Як змінюється опір датчика в сухому ґрунті?
- 4 Який рівень видає модуль, перевищивши порогове значення, задане на компараторі?

## **ПРАКТИЧНА РОБОТА 4**

### **Використання газових сенсорів (датчиків)**

#### **4.1 Мета роботи**

Розроблення програми використання газових сенсорів, що може бути використано в робототехніці.

#### **4.2 Короткі теоретичні відомості**

Датчик газу дає змогу визначати наявність газу в просторі. Існує кілька типів датчиків газу (хімічних сенсорів), в основі роботи яких лежить

принцип перетворення хімічної інформації в електричний сигнал у вигляді зміни частоти, струму або напруги.

Хімічні сенсори:

- електрохімічні;
- оптичні;
- терموкаталітичні;
- п'єзоелектричні;
- напівпровідникові.

Датчик газу MQ-2 (рисунок 4.1) є напівпровідниковим хімічним сенсором типу метал-оксид-напівпровідник (МОП, MOS). Коли газ вступає в контакт із датчиком, опір його чутливого елемента змінюється, тому показання датчика знімають за допомогою резистора, під'єданого як дільник напруги в колі.



Рисунок 4.1 – Датчик газу MQ-2

Робота напруги датчика газу дорівнює 5 В, а споживання – 800 мВт. За допомогою нього можна виявляти концентрації скрапленого нафтового газу, диму, алкоголю, пропану, водню, метану і чадного від 200 до 10000 ppm (мільйонних часток).

Мільйонна частка (ppm) – відношення одного газу до іншого. Іншою поширеною одиницею вимірювання є відсоткова концентрація. Якщо переводити у відсоткову концентрацію, то 10000 ppm = 1 % (10000 частин молекул певного газу на мільйон молекул інших різних газів), коли

1000000 = 100 %. У таблиці 4.1 наведені технічні характеристики датчика газу mq-2.

Таблиця 4.1 - Технічні характеристика датчика газу mq-2

Робоча напруга	5 В
Опір навантаження	20 кОм
Опір нагрівача	33 Ом ± 5 %
Споживана потужність	<800 мВт
Опір чутливості	10 кОм - 60 кОм
Вимірювання концентрації	200 - 10000 ppm
Час розігрівання	понад 180 с

З недоліків датчика газу можна зазначити неможливість визначення, що за газ він визначив. Тому його використовують виключно для визначення змін концентрацій газу.

Зовнішня оболонка – два тонких шари сітки з нержавіючої сталі, які називають «противибуховою сіткою». Вона гарантує безпеку під час пошуку легкозаймистих газів, а також забезпечує захист датчика і відфільтровує зважені частинки, тобто сприймає тільки газоподібні елементи. Шари сітки закріплюють на датчику за допомогою мідного затискного кільця.

На рисунку 4.2 подана структура датчика газу, що складається з шести ніжок, які проходять через чутливий елемент і виходять за межі бакелітової основи. Два виходи (Н) відповідають за нагрівання чутливого елемента і з'єднані через провідний сплав (котушка з нікель-хромового дроту).



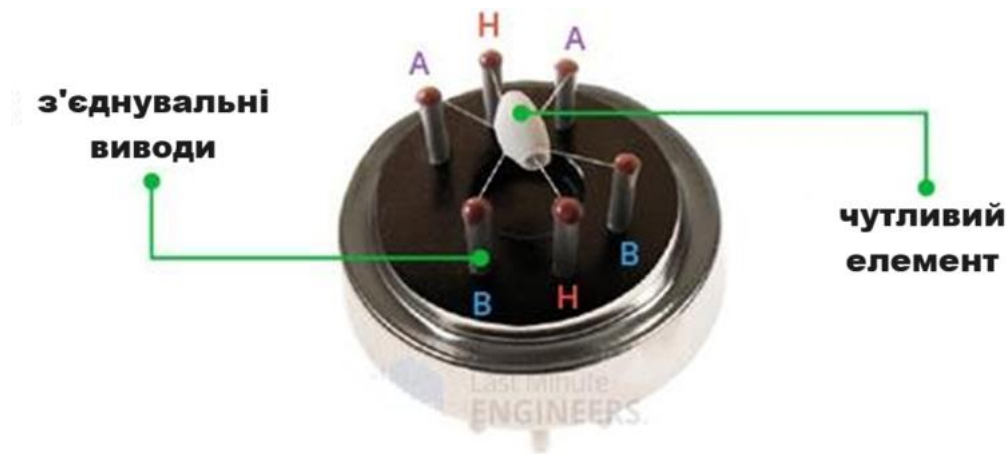


Рисунок 4.2 – Внутрішня структура з чутливим елементом і сполучними виводами

Решта виводів (A і B) відповідають за вихідні сигнали і під'єднані з використанням платинових дротів. Вони відповідають за передавання невеликих змін струму, який проходить через чутливий елемент (рисунок 4.3).

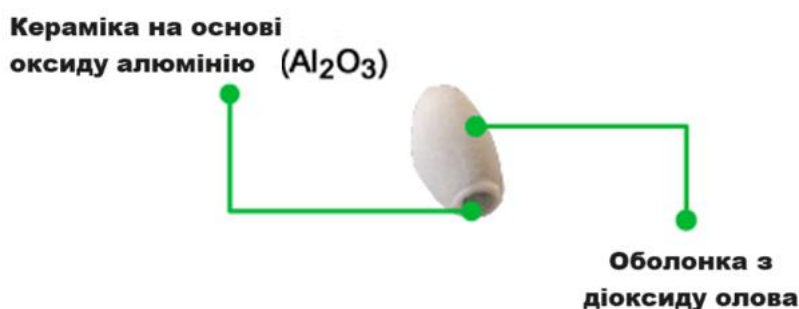


Рисунок 4.3 – Чутливий елемент – кераміка на основі оксиду алюмінію з покриттям із діоксиду олова

Чутливий елемент складається з кераміки оксиду алюмінію (Al<sub>2</sub>O<sub>3</sub>), вкритий діоксидом олова (SnO<sub>2</sub>) (рисунок 4.4). Діоксид олова чутливий до горючих газів, а керамічна підкладка підтримує робочу температуру датчика.

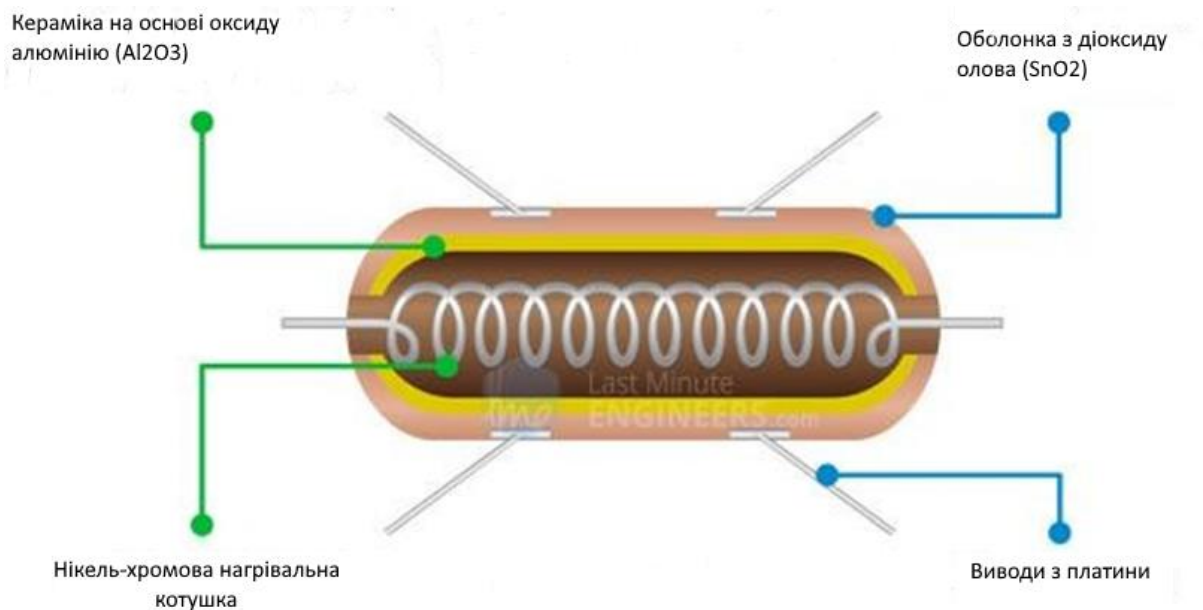


Рисунок 4.4 – Внутрішня структура чутливого елемента датчика газу MQ-2

Принцип роботи такий: коли діоксид олова (частинки напівпровідника) нагрівається на повітрі до високої (робочої) температури, то на його поверхні адсорбується кисень. У чистому повітрі донорні електрони діоксиду олова притягуються до цього кисню, що є свідченням протікання електричного струму. За присутності відновлювальних газів поверхнева густина адсорбованого кисню зменшується під час реакції з відновлювальними газами, через що електрони вивільнюються в діоксид олова, що дає змогу струму вільно протікати через датчик. Напруга збільшується при визначенні датчика газу і падає за його відсутності.

### 4.3 Порядок виконання

Для виконання практичного завдання необхідно виконати низку дій:

- зайти на сайт [www.tinkercad.com](http://www.tinkercad.com);
- вибрати елемент створення кола;
- вибрати такі компоненти: arduino uno, світлодіоди – 2 шт., резистори – 3 шт., п'єзоелемент, датчик газу та макетну плату.

Підключити такі елементи:

а) аноди (+) світлодіодів через обмежувальні резистори – до цифрових пінів arduino, а катоди (-) – до землі (GND) arduino;

б) п'єзоелемент. Плюс (+) підключити до цифрового піна arduino, а мінус (-) – до землі (GND) arduino;

в) датчик газу. Контакт A1 через резистор 10 кОм підключити до землі (GND) arduino, як і контакт H1, контакт A2 – до аналогового піна arduino, контакти B1, H2, B2 – до живлення 5 В arduino.

Написати код програми, що реалізує такі дії: за відсутності газу запалювати один світлодіод (наприклад зелений), п'єзоелемент вимкнений, а в разі визначення газу запалювати інший світлодіод (наприклад червоний), вмикати п'єзоелемент.

```
int redLed = 12;
int greenLed = 11;
int buzzer = 10;
int. smokeA0 = A5;
int sensorThres = 590;
void Setup() {
    pinMode (redLed, OUTPUT);
    pinMode(greenLed, OUTPUT);
    pinMode (buzzer, OUTPUT);
    pinMode (smokeA0, INPUT);
    Serial.begin(9600);
}
void loop() {
    int analogSensor = analogRead(A0);
    Serial.print("Pin A5: ");
    Serial.println(analogSensor);
    if (analogSensor > sensorThres)
    {
        digitalWrite(redLed, HIGH);
        digitalWrite(greenLed, LOW);
        digitalWrite(buzzer, HIGH);
    }
    else
    {
        digitalWrite(redLed, LOW);
        digitalWrite(greenLed, HIGH);
    }
}
```

```
        digitalWrite(buzzer, LOW);
    }
    delay(100);
}
```

У рядках 1-4 вказано піни, до яких підключені елементи. У рядку 5 задано порогове значення АЦП, за якого вважається, що газ є. На рядках 7-10 задано режими роботи пінів. Зчитування значення АЦП з аналогового піна, до якого підключений датчик (рядок 14). На рядках 15, 16 написані значення, отримані з аналогового піна. Якщо значення більше за допустиме, то спалахує червоний світлодіод, вмикається п'єзоелемент (рядки 17-21), інакше (else) спалахує тільки зелений світлодіод, п'єзоелемент вимкнений (рядки 23-27).

### **Запитання для самоконтролю**

- 1 Якого типу хімічний сенсор (датчик газу) поданий у роботі?
- 2 Яке порогове значення АЦП задано, за якого відзначено наявність газу?
- 3 Яке вимірювання концентрацій MQ-2?
- 4 Як змінюється напруга при збільшенні концентрації газу?
- 5 Чи є в датчика можливість визначати, який газ було зафіксовано?

## **ПРАКТИЧНА РОБОТА 5**

### **Використання фоторезисторів (датчиків освітленості)**

#### **5.1 Мета роботи**

Розроблення програми використання фоторезисторів, які можуть бути використані в робототехніці.

## 5.2 Короткі теоретичні відомості

Фоторезистор – напівпровідниковий прилад (датчик), що змінює (зменшує) свій опір у разі зміни освітлення (рисунок 5.1). Чутливим елементом фоторезистора є напівпровідниковий елемент (наприклад сірчистий свинець або сірчистий кадмій), який розташований так, щоб на нього падало світло.

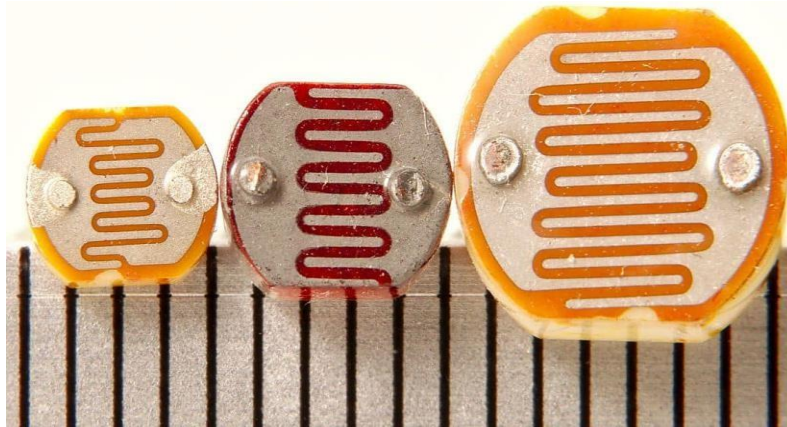


Рисунок 5.1 – Зовнішній вигляд фоторезисторів

На рисунку 5.2 подано конструкцію фоторезистора. Явищем фотопровідності вважають збільшення електропровідності напівпровідника під впливом електромагнітного випромінювання.



Рисунок 5.2 – Конструкція фоторезистора

Принцип роботи фоторезистора. У неактивному стані напівпровідник проявляє властивості діелектрика. Термічний або світловий вплив може служити стимулятором для проведення струму. Під час потрапляння світла на датчик з'являються рухомі носії заряду (електронів) унаслідок поглинання напівпровідником світлової енергії, після чого опір починає зменшуватися (виникає додаткова провідність). Більша чутливість проявляється до інфрачервоних променів, які мають більшу енергію.

На рисунку 5.3 подано схему з'єднання датчика з резистором, під'єднаним як дільник напруги. Зроблено це для контролю вхідної напруги і зняття показань із датчика.

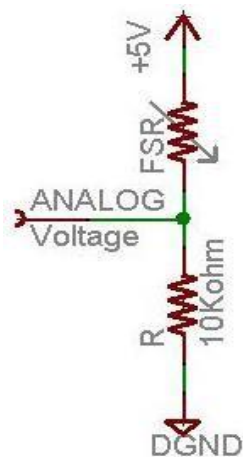


Рисунок 5.3 – Принципова схема підключення фоторезистора

### Порядок виконання

Для виконання практичного завдання потрібно зайти на сайт [www.tinkercad.com](http://www.tinkercad.com), потім вибрати створення кола. Можливості програми (сайту) наведено в додатку А. Вибрати такі компоненти: arduino uno, світлодіоди – 5 шт., резистори – 6 шт., п'єзоелемент, фоторезистор і макетну плату.

Підключити такі елементи:

– аноди п'яти світлодіодів через обмежувальні резистори – до п'яти цифрових пінів arduino, а катоди – до землі (GND) arduino;

– п'єзоелемент. Плюс (+) підключити до цифрового піна arduino, а мінус – до землі (GND) arduino;

– резистивний датчик тиску не має полярності, тому неважливо, який контакт буде прийнятий за плюс (+), а який – за мінус (-). Контакт, прийнятий за плюс (+), під'єднати до живлення 5В arduino, а контакт, прийнятий за мінус, через знижувальний резистор 10 кОм – до землі (він виступатиме як дільник напруги). Потім точку кола між резистором і фоторезистором (змінним резистором) під'єднати до аналогового піна для зняття показань із датчика.

Написати скетч, що реалізує такі дії: у разі освітленості на датчик, запалювати послідовно світлодіоди, починаючи з третього світлодіода ввімкнути п'єзоелемент і залишити його ввімкненим до останнього, п'ятого, світлодіода. Також цю схему можна зібрати в очному форматі, використовуючи arduino uno і кабель для завантаження скетчу.

На рядках 3-9 задаємо режими роботи пінів, до яких підключені елементи. Значення резистивного датчика освітленості зчитуємо з аналогового піна A0 (рядок 13).

Лістинг програми (скетч):

```
int h = 0;
void setup () {
  pinMode(A0, INPUT);
  pinMode(13, OUTPUT);
  pinMode(12, OUTPUT);
  pinMode(11, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(9, OUTPUT);
  pinMode(8, OUTPUT);
  Serial.begin (9600);
}
void loop() {
  h = analogRead(A0);
  if(h >100) {
```

```

        digitalWrite(13, HIGH);
        digitalWrite (8, LOW) ;
    }else{
        digitalWrite(13, LOW);
        digitalWrite(8, LOW);
    }
    if(h >200) {
        digitalWrite(12, HIGH);
        digitalWrite(8, LOW);
    }else{
        digitalWrite (12, LOW);
        digitalWrite(8, LOW);
    }
    if(h >300) {
        digitalWrite(11, HIGH);
        digitalWrite(8, HIGH);
    }else{
        digitalWrite(11, LOW);
        digitalWrite(8, LOW);
    }
    if(h >400) {
        digitalWrite(10, HIGH);
        digitalWrite(8, HIGH);
    }else{
        digitalWrite(10, LOW);
        digitalWrite(8, LOW);
    }
    if(h >450) {
        digitalWrite(9, HIGH);
        digitalWrite(8, HIGH);
    }else{
        digitalWrite(9, LOW);
        digitalWrite(8, LOW);
    }
    Serial.println(h);
    delay(1000);
}

```

За значень вище 100 подається високий сигнал на 13 пін (HIGH), на другий пін подається низький сигнал (LOW), інакше (else) подається на восьмий і тринадцятий піни низький сигнал (LOW) (14-20 рядки). За значень вище 200 подається високий сигнал на тринадцятий пін (HIGH), на восьмий



пін подається низький сигнал (LOW), інакше (else) подається на восьмий і тринадцятий піни низький сигнал (LOW) (21-27 рядки). За значень вище 300 подається високий сигнал на восьмий і одинадцятий піни (HIGH), інакше (else) подається на восьмий і одинадцятий піни низький сигнал (LOW) (28-34 рядки). За значень вище 400 подається високий сигнал (HIGH) на восьмий і десятій піни, інакше (else) подається на восьмий і десятій піни низький сигнал (LOW) (35-41 рядки). За значень вище 450 подається високий сигнал (HIGH) на восьмий і дев'ятий піни, інакше (else) подається на восьмий і дев'ятий піни низький сигнал (LOW) (42-47 рядки). Виведення значень "h" на монітор порту й очікування 1000 мс (рядки 49, 50).

Запустити моделювання та перевірити роботу фоторезистора.

### **Запитання для самоперевірки**

1 Як змінюється напруга фоторезистора під час впливу на нього світлового випромінювання?

2 Як змінюється опір фоторезистора під час впливу на нього світлового випромінювання?

3 Якому кольору відповідає пік чутливості досліджуваних фоторезисторів?

## СПИСОК ЛІТЕРАТУРИ

- 1 Цвіркун Л. І., Грулер Г. Робототехніка та мехатроніка: навч. посіб. / за заг. ред. Л. І. Цвіркуна ; М-во освіти і науки України, Нац. гірн. ун-т. Вид. 3-тє, перероб. і доп. Дніпро: НГУ, 2017. 224 с.
- 2 Документація до програмного забезпечення ABB Robotic Studio. URL: <http://developercenter.robotstudio.com/>.
- 3 Офіційний сайт розробників Gazebo. URL: <http://gazebosim.org>.
- 4 Офіційний сайт розробників V-REP. URL: <http://coppeliarobotics.com/>.
- 5 Документація відкритої бібліотеки планування руху. URL: <http://ompl.kavrakilab.org>.
- 6 Документація для розробників мовою Lua. URL: <http://www.lua.ru/doc>.
- 7 Список регулярних функцій V-REP з докладним описом. URL: <http://www.coppeliarobotics.com/helpFiles/en/apiFunctionListCategory.htm>.
- 8 Методичний посібник з додержання вимог нормоконтролю у студентській навчальній звітності. Студентська навчальна звітність. Текстова частина (пояснювальна записка). Загальні вимоги до побудови, викладення та оформлення /за заг. ред. Л. М. Козара. Харків : УкрДАЗТ, 2014. 58 с.
- 9 Tinkercad arduino – Найкращий онлайн симулятор ардуіно. URL: <https://arduinomaster.com/program/simulyator-arduino-tinkercad-circuits/>.

## ДОДАТОК А

### Основи роботи в Tinkercad

#### *Система Tinkercad*

Tinkercad – це безкоштовна онлайн-програма для 3D-моделювання, яка запускається у веббраузері. Розділ «Схеми» у Tinkercad являє собою симулятор електронної схеми з Arduino Uno або мікророзрядною платою чи чипом ATtiny у браузері. Код може бути створений за допомогою CodeBlocks, що являють собою графічні фрагменти коду, які можна зібрати разом, переміщаючи курсором миші. Також можливе програмування з використанням тексту коду. Схема може бути зібрана з компонентів, але є «Стартери» – повні схеми з кодом.

До складу Tinkercad входить бібліотека сервоприводів Arduino. Неможливо вибрати або завантажити інші бібліотеки. Схема може мати аналогові компоненти, які повністю моделюються.

Хоча Tinkercad є простим вступом у програмування та електроніку, у ньому наявні функції для досвідчених користувачів: одночасно можна моделювати систему з плат, наприклад кілька плат Arduino, взаємопов'язаних одна з одною.

#### *Датчики в Tinkercad*

Нижче подано види датчиків, які є в Tinkercad (рисунок А.1).



Рисунок А.1 – Датчики в Tinkercad

## Arduino та програмування в Tinkercad

У системі Tinkercad є можливість спільної роботи кіл із платою Arduino Uno. В її основі лежить чип ATmega – в останній ревізії Arduino Uno R3 – це ATmega328. Її характеристики та рисунок наведено нижче (таблиця А.1, рисунок А.2).

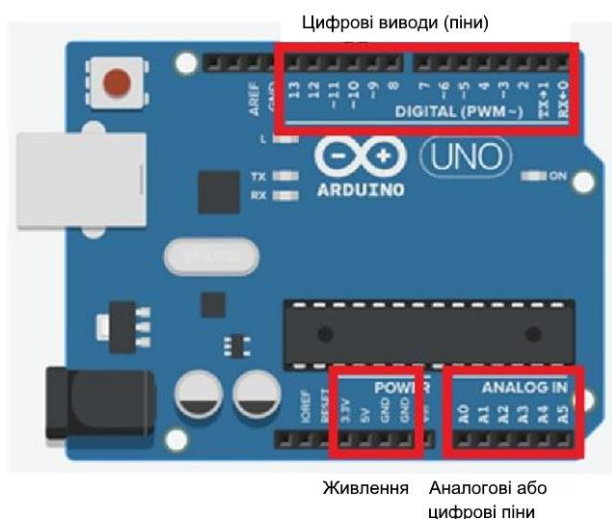


Рисунок А.2 – Arduino Uno в Tinkercad

Таблиця А.1 – Характеристики Arduino Uno

Мікроконтролер	ATmega328
Робоча напруга	5 В
Напруга живлення (рекомендована)	7-12 В
Напруга живлення (гранична)	6-20 В
Цифрові входи/виходи	14 (із них шість можна використовувати як ШІМ-виходи)
Аналогові входи	6
Максимальний струм одного виводу	20 мА
Максимальний вихідний струм виводу 3.3V	50 мА
Flash-пам'ять	32 КБ (ATmega328), з яких 0.5 КБ використовує завантажувач
SRAM	2 КБ (ATmega328)
EEPROM	1 КБ (ATmega328)
Тактова частота	16 МГц

Плата arduino uno має шість аналогових пінів, позначених від A0 до A5 (інші плати можуть мати різну кількість пінів), робота з напругою від 0 до 5 В. Завдяки АЦП аналогові піни зчитують напругу, що подається на них. Arduino uno має мікроконтролери Atmega328, які містять шестиканальний АЦП, роздільна здатність становить 10 біт, що дає змогу отримувати на виході значення в 1024 градації (від 0 до 1023).

AnalogRead() – вбудований метод, що дає змогу зчитувати показання напруги в градаціях від 0 до 1024. При цьому для зворотної конвертації в напругу потрібно провести множення отриманої градації на величину однієї градації за напругою, яка визначається як  $5/1024 = 4,88$  мВ. Ця величина і є мінімальною точністю визначення напруги для Arduino Uno.

Також плата має 13 цифрових пінів, що позначаються від D0 до D13.

*Режими роботи пінів.* Цифровий пін може перебувати як у стані виходу, так і стані входу. У режимі входу зчитується напруга, а в стані виходу – видає таку саму напругу. Функція pinMode(pin, mode) задає режим роботи, де pin – номер піна, а mode – режим:

- INPUT – вхід;
- OUTPUT – вихід;
- INPUT\_PULLUP – підтягнутий до живлення вхід.

Якщо в режимі входу пін мікроконтролера не під'єднаний нікуди, то ловить із повітря різні наводки, отримуючи шумові випадкові значення. Для завдання піну конкретного режиму використовують підтяжку резистором до землі або живлення. За допомогою внутрішнього резистора режим INPUT\_PULLUP вмикає вбудовану в мікроконтролер підтяжку піна до живлення.

*Виведення цифрового сигналу.* Цифровий пін у режимі виходу генерує цифровий сигнал, тобто видає напругу. Поняття «цифровий» пов'язане з двома станами, 0 і 1, тобто сигналами низького і високого рівня:

- сигнал низького рівня (0 В) – під'єднання до землі мікроконтролера;

– сигнал високого рівня під'єднується до живлення мікроконтролера. Сигнал високого рівня залежить від тієї напруги, якою живиться плата. У разі живлення безпосередньо від джерела 5 В на піні і буде 5 В, при живленні від USB з втратою на захисному діоді значення дорівнює близько 4,7 В.

Функція `digitalWrite(pin, value)` служить для подавання цифрового сигналу:

- `pin` – пін GPIO (нумерація вище);
- `value` – рівень сигналу: високий (HIGH або 1), низький (LOW або 0).

Функцію `pinMode(10, OUTPUT)` використовують для виведення сигналу при переведенні пін в режим OUTPUT.

### ***Порядок роботи***

У більшості випадків для роботи з проєктами Arduino застосовують такий алгоритм дій: [9]

- створюємо нову схему або відкриваємо наявну для редагування;
- використовуючи візуальний редактор, створюємо схему (у нашому випадку з включенням плати Arduino Uno);
- готуємо скетч у редакторі коду і завантажуюмо його у віртуальний контролер;
- запускаємо режим симуляції, за якої плата віртуально підключається до джерела живлення, і схема починає працювати. Вносимо початкові дані для датчиків і спостерігаємо реакцію схеми як візуально, так і на віртуальному моніторі порту всередині самого сервісу.

### ***Перший крок. Створюємо схему Circuit***

Переходимо в [www.tinkercad.com](http://www.tinkercad.com), розділ *Circuit*, і натискаємо на кнопку Create, вибираючи тип – Circuit. Після цього кроку відкривається візуальне середовище редагування, у якому ми зможемо як накреслити схему, так і написати та налагодити скетч arduino.

### ***Другий крок. Підготовка електронної схеми***

Створюючи схему, виконуємо такий порядок дій:

- вибираємо потрібні компоненти з бібліотеки компонентів внизу екрана і розміщуємо їх у полі редактора;
- з'єднуємо компоненти за допомогою віртуальних провідників, рисуємо їх мишкою;
- редагуємо параметри компонентів (наприклад величину опору в резисторів або колір проводів).

Операція вибору з бібліотеки досить проста. Список елементів знаходиться внизу. Вибравши елемент, ми натискаємо на ньому, потім переміщуємо в потрібне місце на схемі і натискаємо повторно. Вікно зі списком компонентів можна приховати або показати, натискаючи на перемикач «Components» у панелі інструментів.

Для роботи нам доступно безліч уже готових елементів, від резистора і батарейки до модулів Arduino. Для зручності навігації всі елементи розбиті на три вкладки:

- Basic Components – основні компоненти;
- Allcomponents – усі доступні компоненти;
- Starters – готові попередньо встановлені схеми.

Найцікавішою для нас зараз є третя закладка – Starters. Творці сервісу підготували кілька готових схем, які можна одразу ж довантажити в проєкт і редагувати на свій розсуд.

Знайдіть у списку будь-яку схему з Arduino і натисніть на неї. Після повторного натискання елементи схеми будуть розміщені в області редагування. Давайте для прикладу виберемо схему триточкового музичного інструмента (рисунок А.3).

Якщо схема не влізла в екран, виконайте масштабування (натисніть на кнопку масштабу на панелі інструментів).

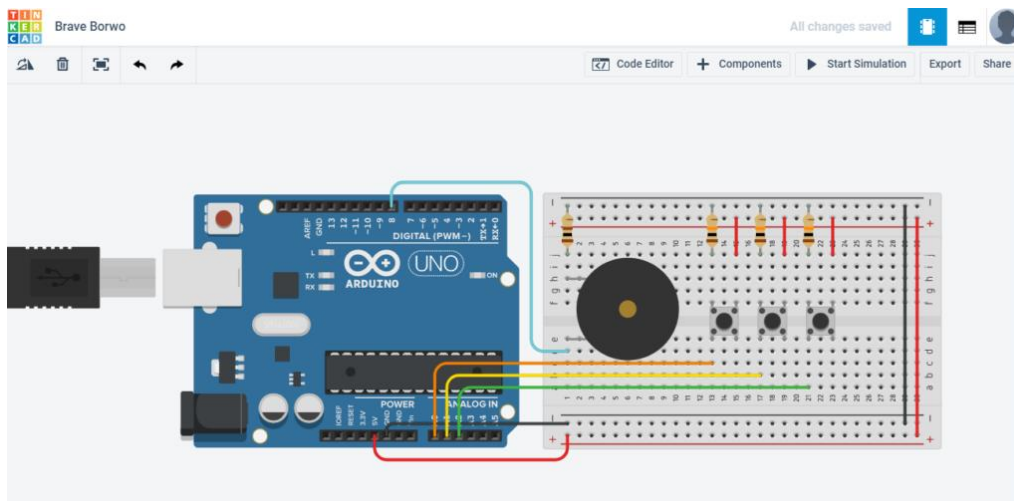


Рисунок А.3

Натиснувши на роз'єм ардуіно або ніжки електронних компонентів, можна «припаяти» до неї дрiт, який натисканнями мишки ми протягуємо по всій нашій платі до бажаної точки (рисунок А.4).

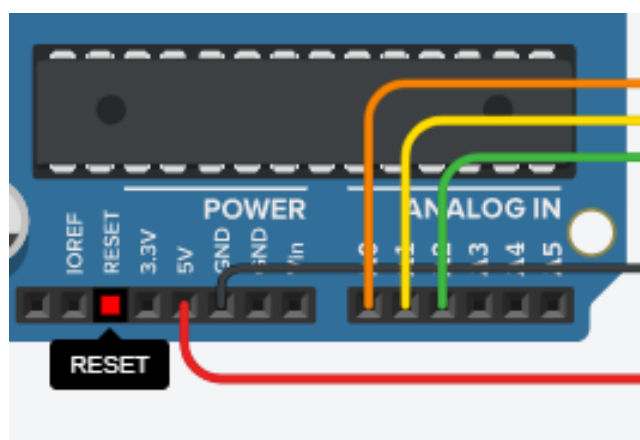


Рисунок А.4

Кути дроту красиво заокруглюються, є можливість вирівнювати дрiт по вертикалі або горизонталі (поява синіх ліній підкаже нам вертикаль і горизонталь відповідно). Для скасування встановлення дроту потрібно натиснути на Esc або мишкою натиснути на відповідну іконку на панелі інструментів.



Натиснувши на компонент, ми можемо відредагувати його властивості (рисунок А.5).

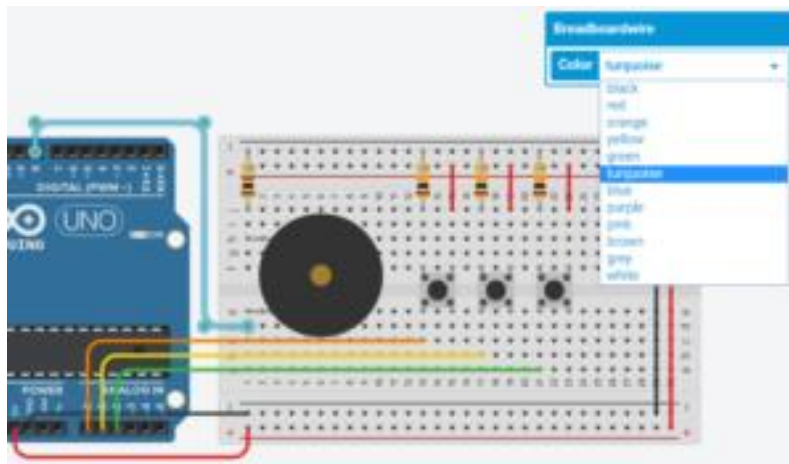


Рисунок А.5

*Третій крок. Програмуємо скетч віртуального Arduino*

Усі інструменти для редагування коду стають доступні після переходу у відповідний режим при натисканні на кнопку «Code Editor» у верхній панелі.

У режимі редагування коду нам доступні такі варіанти дій:

- завантаження скетчу у «віртуальний контролер» і запускання симулятора;
- перемикання у візуальний редактор коду типу Scratch;
- перемикання в текстовий редактор коду;
- підключення бібліотек;
- завантаження коду на свій комп'ютер у вигляді файлу з розширенням .ino (скетч arduino);
- запускання налаштування з можливістю створення точок зупинок і моніторингом станів змінних;
- відображення або приховування вікна монітора (рисунок А.6).

```

1  /*
2  Keyboard
3
4  Plays a pitch that changes based on a changing
5  input circuit:
6  * 3 pushbuttons from +5V to analog in 0 through
7  3
8  * 3 10K resistors from analog in 0 through 3 to
9  ground
10 * 8-ohm speaker on digital pin 8
11 */
12
13 int pos = 0;
14
15 void setup()
16 {
17   pinMode(A0, INPUT);
18   pinMode(8, OUTPUT);
19   pinMode(A1, INPUT);
20   pinMode(A2, INPUT);
21 }
22
23 void loop()
24 {
25   // if button press on A0 is detected
26   if (digitalRead(A0) == HIGH) {
27     tone(8, 440, 100); // play tone 57 (A4 = 440 Hz)
28   }
29   // if button press on A1 is detected
30   if (digitalRead(A1) == HIGH) {
31     tone(8, 494, 100); // play tone 59 (B4 = 494 Hz)
32   }
33   // if button press on A2 is detected
34 }

```

Libraries with compatible components  
Libraries that interact with components available in the editor.

EEPROM	Reading and writing to "permanent" storage	<a href="#">Include</a>
IRremoteInterruptions	Interrupt based library to decode IR sensors	<a href="#">Include</a>
IRremote	Library to decode IR sensors	<a href="#">Include</a>
LiquidCrystal	Controlling liquid crystal displays (LCDs)	<a href="#">Include</a>
Keypad	Allows reading keypad button pushes	<a href="#">Include</a>
NeoPixel	Controlling NeoPixel LEDs	<a href="#">Include</a>
Servo	Controlling servo motors	<a href="#">Include</a>
Software Serial	Allow serial communication on other digital pins of the Arduino	<a href="#">Include</a>
Wire	This library allows you to communicate with I2C / TWI devices	<a href="#">Include</a>

Рисунок А.6

По суті перед нами повноцінне середовище розроблення, що має нехай і досить скромний, але цілком достатній для більшості випадків набір інструментів. А наявність в одному середовищі візуального режиму і механізмів налаштування робить цей сервіс по-справжньому унікальним і вкрай зручним для новачків.

#### *Четвертий крок. Запускаємо симулятор arduino*

Є два способи запуску симулятора. Перший – натиснути на кнопку «Start Simulation» у верхній панелі (рисунок А.7). Другий – використовувати кнопку Upload&Run у режимі редагування коду.

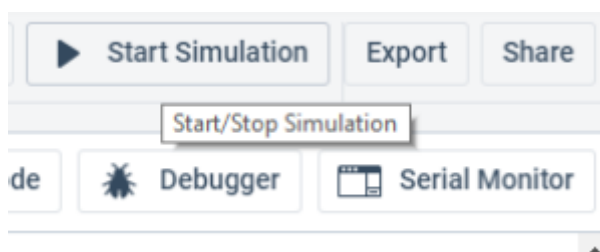


Рисунок А.7

В обох випадках для зупинки роботи симулятора потрібно просто ще раз натиснути на верхню кнопку (у режимі симуляції напис зміниться на «Stop Simulation»).

Що відбувається під час симуляції? А практично те саме, що і під час під'єднання живлення до реальної схеми. Лампочки горять, з п'єзовипромінювача чути звуки, двигуни крутяться. Ми можемо відстежувати поточні показники (напруга, струм) за допомогою інструментів моніторингу. А можемо самі створювати зовнішні сигнали, подаючи на датчики необхідні значення і відстежувати потім реакцію програми. Наприклад, можна задати мишкою розташування об'єкта до датчика відстані, значення освітленості для фоторезистора, повернути ручку потенціометра. Також чудово працюють такі елементи, як LCD дисплей – ми побачимо виведену інформацію просто на екрані візуального компонента (рисунок А.8).

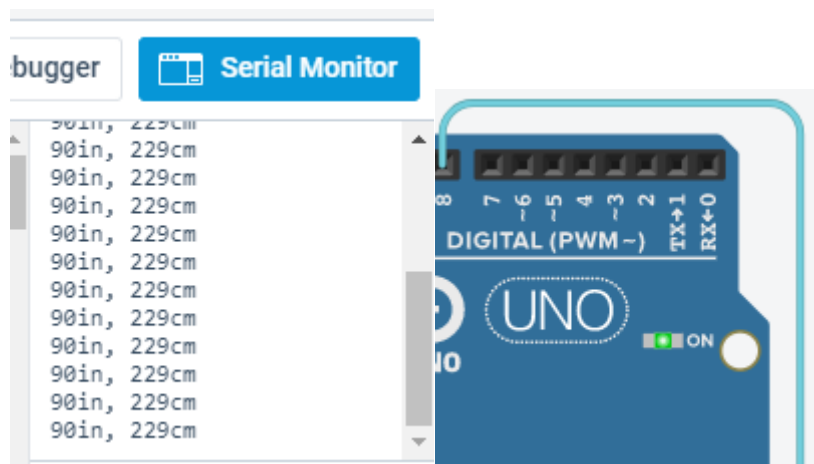
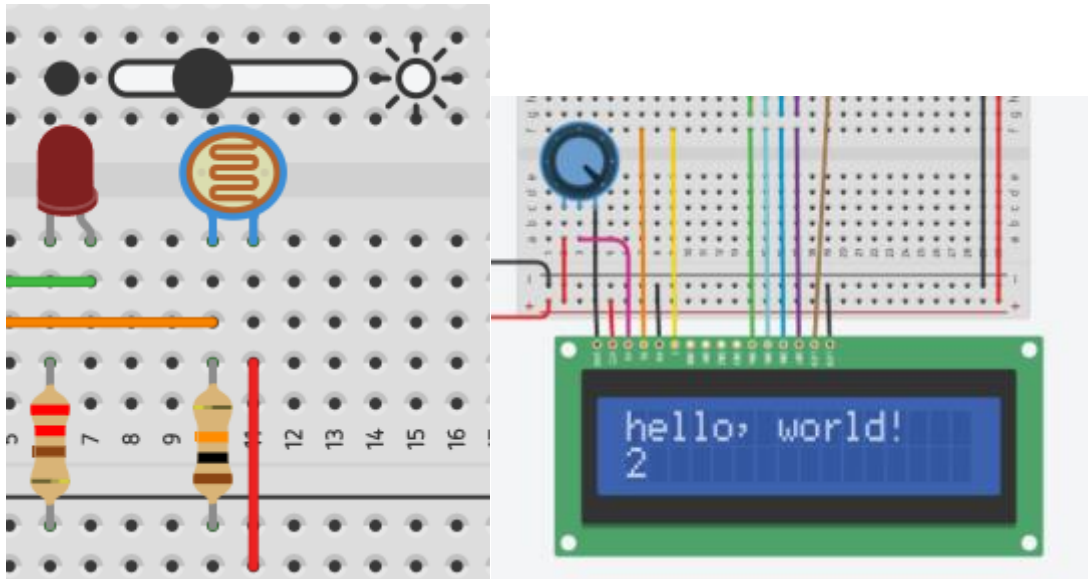
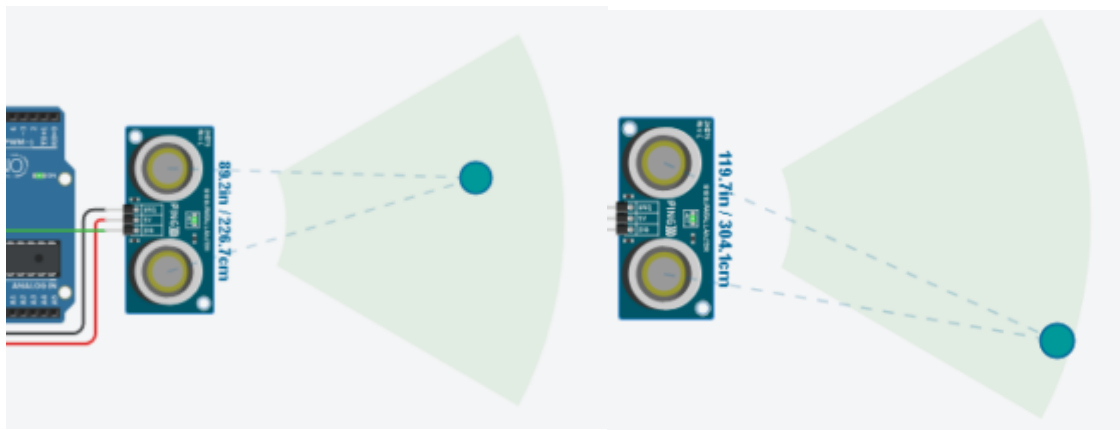


Рисунок А.8

*Читання цифрового сигналу.* Цифровий пін має можливість повідомляти тільки про відсутність напруги (0 або LOW) або її наявність (1 або HIGH), проміжки напруги зчитуються від 0 до  $V_{CC}/2$  В і від  $V_{CC}/2$  до

$V_{CC}$  мікроконтролера. Мікроконтролер не дає змогу подавати на нього напруги, вищі за передбачені.

Функцію `digitalRead(pin)` використовують для читання рівня сигналу. Для читання сигналу за допомогою `pinMode()` пін потрібно перевести в режим INPUT. Наступний код виводитиме в порт сигнал на піні D5. Якщо під'єднати його проводом до  $V_{CC}$ , отримаємо 1, якщо до GND – отримаємо 0.

```
void setup() {  
    Serial.begin(9600);  
}  
void loop(){  
    Serial.println(digitalRead(5));  
}
```

#### *Контактні перетворювачі. Кнопка*

Кнопка (кнопковий перемикач) є найпростішим пристроєм, функцією якого є замикання і розмикання контакту. Цей датчик параметричного типу.

Кнопки поділяються на такі типи:

- з фіксацією, тобто залишається затиснутою після відпускання і без фіксації – відключається назад;
- нормально розімкнена – під час натискання замикає контакти, і нормально замкнена, яка під час натискання розмикає контакти;
- тактові кнопки – замикають або розмикають контакт, у яких зазвичай ноги з'єднані вздовж через корпус (рисунок А.9). Перемикачі мають три контакти: загальний COM, нормально відкритий NO і нормально закритий NC (рисунок А.10).

Для виконання практичного завдання потрібно зайти на сайт [www.tinkercad.com](http://www.tinkercad.com), потім вибрати створення кола. Обрати такі компоненти: arduino uno, світлодіод, резистор – 2 шт., кнопку та макетну плату (за бажанням).

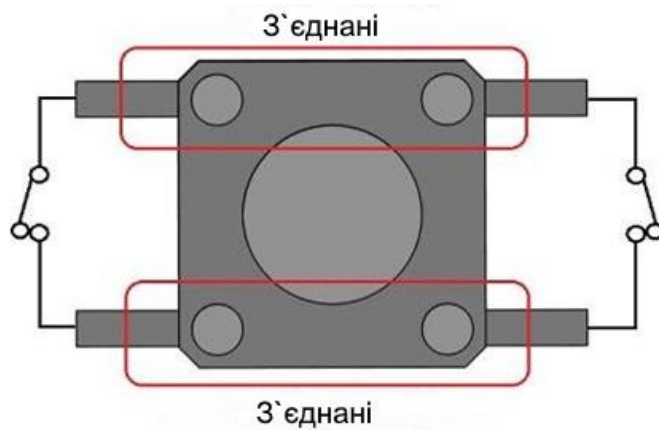


Рисунок А.9 – Тактова кнопка

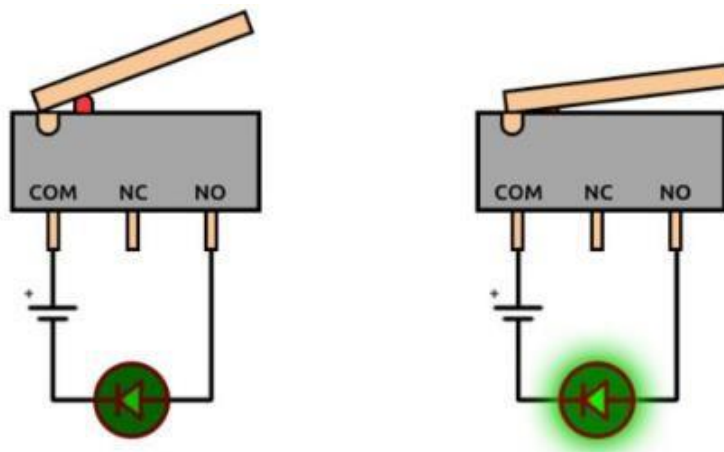


Рисунок А.10 – Перемикачі

Підключити такі елементи:

- світлодіод. Анод (плюс) світлодіода підключити через обмежувальний резистор до цифрового піна arduino, катод (мінус) – до землі arduino (GND);

- кнопка. Одну ногу першої пари під'єднати до живлення 5 В arduino, а ноги другої пари – до цифрового піна arduino і через знижувальний резистор 10 кОм до землі arduino (GND).

Також зібрати експериментальну схему в очному форматі. Для цього знадобиться програма arduino і кабель, що підключається до arduino uno, за

допомогою чого можна буде завантажувати код у плату. У цьому разі замість тактової кнопки буде дано її модуль, який має три контакти. Підключення здійснюватиметься так: мінус (-) під'єднати до землі arduino (GND), сигнал (S) – до цифрового піна arduino, а плюс – до живлення 5 В arduino.

Написати два програмні коди, що реалізують такі дії:

а) під час натискання і відпускання кнопки світлодіод відповідним чином загоряється і гасне. На рядках 1 і 2 вказано піни, до яких було під'єднано кнопку і світлодіод. На рядках 4, 5 задано режими роботи пінів, де пін із кнопкою вхідний, а пін зі світлодіодом вихідний. Наступною дією буде увімкнення світлодіода в разі натиснутої кнопки, тобто якщо кнопка натиснута (рядок 8), то запалюємо світлодіод (рядок 9). Інакше (else), вимикаємо світлодіод (рядок 12);

```
1 int switchPin = 2;
2 int ledPin = 3;
3 void setup() {
4   pinMode(switchPin, INPUT);
5   pinMode(ledPin, OUTPUT);
6 }
7 void loop() {
8   if(digitalRead(switchPin) == HIGH) {
9     digitalWrite(ledPin, HIGH);
10  }
11  else {
12    digitalWrite(ledPin, LOW);
13  }
14 }
```

б) під час натискання і відпускання кнопки світлодіод загоряється, а гасне вже під час наступного натискання і відпускання кнопки. Важливо використовувати програмний метод затримки для уникнення деренчання, додаючи в скетчі паузу 10-50 мс між під'єднанням значень з піна arduino.

```

1  int switchPin = 2;
2  int ledPin = 3;
3  boolean lastButton = LOW;
4  boolean currentButton = LOW;
5  boolean ledOn = false;
6  void setup() {
7  pinMode(switchPin, INPUT);
8  pinMode(ledPin, OUTPUT);
9  }
10 boolean debounce(boolean last) {
11 boolean current = digitalRead(switchPin);
12 if(last != current) {
13 delay(5); current = digitalRead(switchPin);
14 }
15 return current;
16 }
17 void loop() {
18 currentButton = debounce(lastButton);
19 if(lastButton == LOW && currentButton == HIGH) {
20 ledOn = !ledOn;
21 }
22 lastButton = currentButton;
23 digitalWrite(ledPin, ledOn);
24 }

```

На рядках 1, 2 вказують піни, до яких під'єднано кнопку і світлодіод. На рядках 3, 4 вказують попередній і поточний стан кнопки, а рядку 5 – поточний стан світлодіода. На рядках 7, 8 вказують режим роботи пінів: для кнопки вхід, світлодіода вихід. Робота з програмним усуненням деренчання: отримуємо минулий стан кнопки. Факт натискання визначається за допомогою функції `digitalRead()`. Якщо поточний стан відрізняється від попереднього (рядок 12), то чекаємо 5 мс і присвоюємо поточний стан кнопки (рядок 13). Передаємо функції деренчання значення за замовчуванням (LOW) (рядок 18). Потім потрібно перевірити, відпустили кнопку чи ні (рядок 19), якщо вона була натиснута, то інвертуємо значення змінної стану світлодіода (рядок 20). Присвоюємо минулий стан кнопки (рядок 22) для подальшого визначення натискання кнопки і змінюємо стан світлодіода (рядок 23).



МОДЕЛЮВАННЯ ТА РОБОТА З ДАТЧИКАМИ  
І ВИКОНАВЧИМИ ПРИСТРОЯМИ РОБОТОТЕХНІКИ

МЕТОДИЧНІ ВКАЗІВКИ

до практичних робіт і самостійної роботи

з дисципліни

*«ТЕОРЕТИЧНІ ОСНОВИ РОБОТОТЕХНІКИ»*

Відповідальний за випуск Щєбликіна О. В.

Редактор Ібрагімова Н. В.

---

Підписано до друку 03.04.2024 р.  
Умовн. друк. арк. 3,0. Тираж . Замовлення № .  
Видавець та виготовлювач Український державний університет залізничного  
транспорту,  
61050, Харків-50, майдан Фейєрбаха,7.  
Свідоцтво суб'єкта видавничої справи ДК № 6100 від 21.03.2018 р.