

## **Практична робота 11. Робота з ультразвуковим далекоміром HC-SR04**

### **Теоретичні відомості**

Далекомір – це пристрій для вимірювання відстані до деякого предмету. Далекімір використовується в різних ситуаціях. Простий колісний робот може використовувати цей прилад для виявлення перешкод. Літаючий дрон використовує далекомір для польоту над землею на заданій висоті. За допомогою далекоміра можна навіть побудувати карту приміщення, застосувавши спеціальний алгоритм SLAM.

#### **Принцип роботи далекомірів**

Розглянемо принцип роботи одного з найпопулярніших датчиків – ультразвукового (УЗ) далекоміра. Існує багато різних модифікацій подібних пристроїв, але всі вони працюють за принципом вимірювання часу проходження відбитого звуку. Тобто датчик відправляє звуковий сигнал в заданому напрямку, потім ловить відбите відлуння і обчислює час польоту звуку від датчика до перешкоди і назад (рис. 11.1).

Швидкість звуку в деякому середовищі величина постійна. Знаючи швидкість звуку в повітрі і час польоту звуку до перешкоди, можна розрахувати пройдену звуком відстань за формулою:

$$S = v \cdot t$$

де  $v$  – швидкість звуку в м/с, а  $t$  – час в секундах.



Рисунок 11.1 – Принцип роботи ультразвукового далекоміра

Щоб впоратися зі своїм завданням, далекомір має дві важливі конструктивні особливості. По-перше, щоб звук добре відбивався від перешкод, датчик випускає ультразвук з частотою 40 кГц. Для цього в датчику є п'єзокерамічний випромінювач, який здатний генерувати звук такої високої частоти. По-друге, випромінювач влаштований таким чином, що звук поширюється не на всі боки (як це буває у звичайних динаміків), а у вузькому

напрямку. На рисунку 11.2 представлена діаграма спрямованості типового УЗ далекоміра.

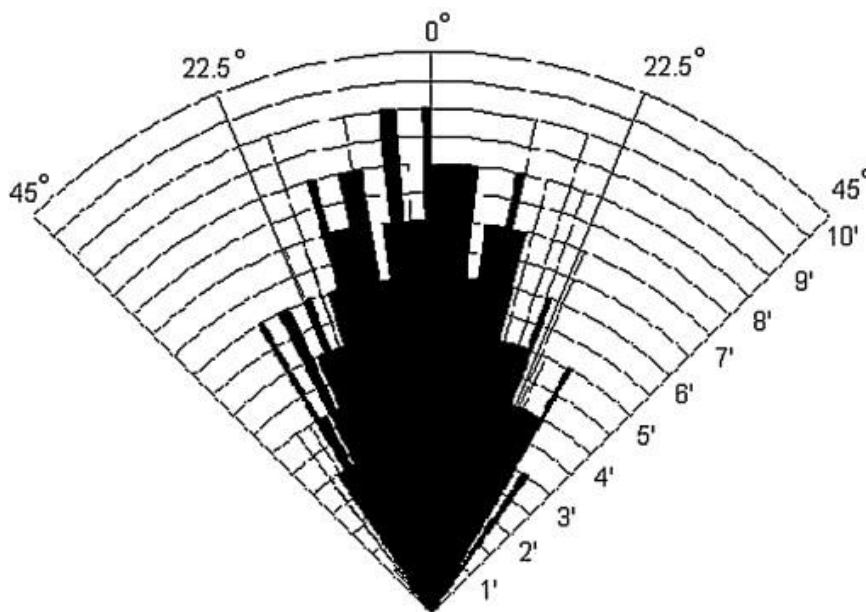


Рисунок 11.2 – Діаграма спрямованості звуку далекоміра

Як видно з діаграми, кут огляду самого простого УЗ далекоміра становить приблизно 50-60 градусів. Для типового варіанту використання, коли датчик визначає перешкоди перед собою, такий кут огляду цілком прийнятний.

Також варто зазначити два серйозні недоліки УЗ далекоміра. Перший полягає в тому, що поверхні, які мають пористу структуру добре поглинають ультразвук, і датчик не може виміряти відстань до них. Такі ж проблеми виникають при вимірюванні відстані до стіни покритої поролоном. Другий недолік пов'язаний зі швидкістю звукової хвилі. Ця швидкість недостатньо висока, щоб вимірювати відстань до об'єктів, що швидко рухаються.

### **Ультразвуковий далекомір HC-SR04**

Далекомір HC-SR04 (рис. 11.3) є найдешевшим далекоміром для використання в любительських проектах. При малій ціні має непогані характеристики, здатний вимірювати відстань в діапазоні від 2 до 450см з точністю 0.5 – 1 см.

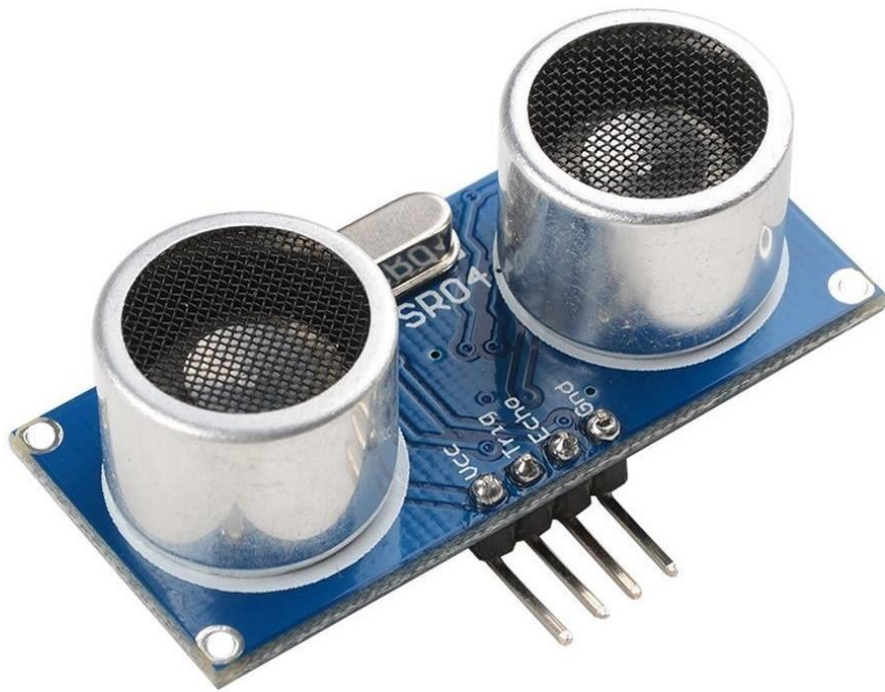


Рисунок 11.3 – Ультразвуковий далекомір HC-SR04

Принцип роботи датчика можна умовно розділити на 4 етапи:

1. Подача імпульсу тривалістю 10 мкс, на вихід Trig.
2. Всередині далекоміра вхідний імпульс перетворюється в 8 імпульсів частотою 40 кГц і надсилається вперед через передавач.
3. Дійшовши до перешкоди, послані імпульси відбиваються і потрапляють на приймач. Отримується вихідний сигнал на виході Echo.
4. Безпосередньо на стороні контролера отриманий сигнал переводиться у відстань за формулою:

$$\text{ширина імпульсу (мкс)} / 58 = \text{відстань (см)}$$

Модуль підключається за допомогою чотирьох провідників. Контакти VCC та GND служать для підключення живлення, а Trig та Echo – для відправки та прийому сигналів далекоміра, підключивши їх, наприклад до 2 та 3 пінів плати Arduino (рис. 11.4).

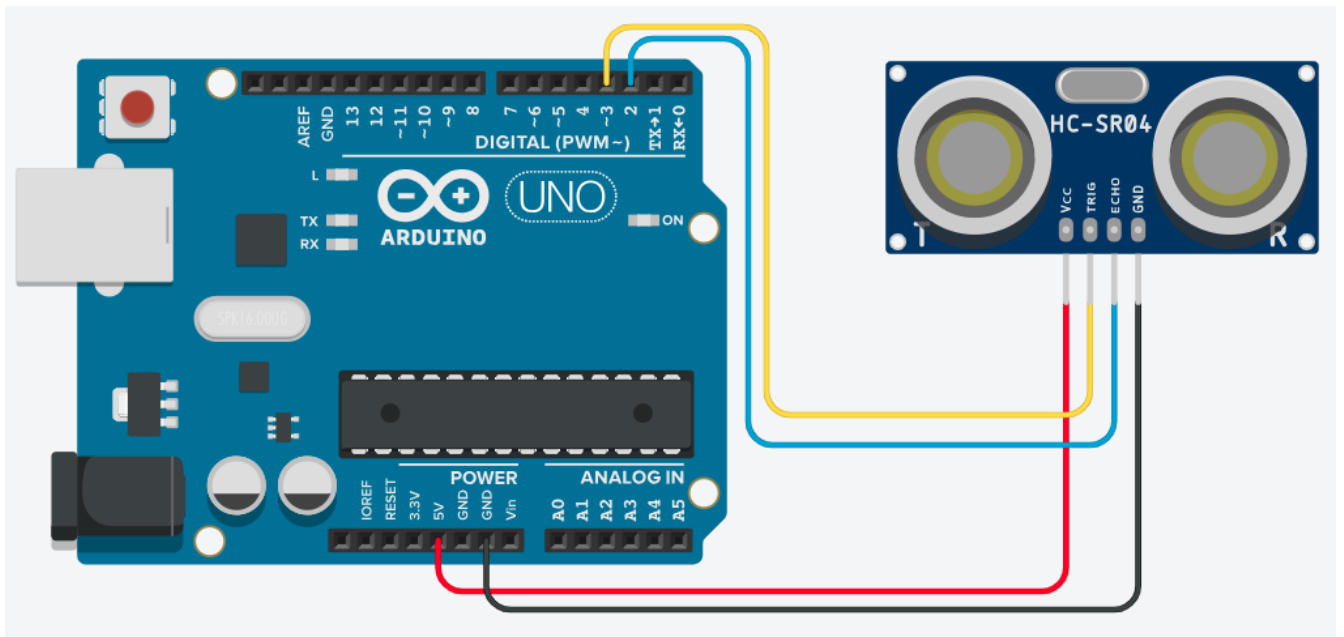


Рисунок 11.4 – Схема підключення ультразвукового далекоміра HC-SR04 до Arduino Uno

Код скетчу для отримання даних з далекоміра та передача їх через послідовний інтерфейс представлений в лістингу 11.1:

Лістинг 6.1 – Отримання відстані до об'єкту за допомогою далекоміра

```
int echoPin = 2; // пін Echo дальноміра
int trigPin = 3; // пін Trig дальноміра
int duration, cm; // змінні для проміжних обрахунків

void setup()
{
  Serial.begin(9600); // встановлення пслідового з'єднання
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
}

void loop()
{
  digitalWrite(trigPin, HIGH); // початок передачі імпульсу
  delayMicroseconds(10); // затримка в 40 мкс
```

```
digitalWrite(trigPin, LOW);           // завершення передачі
duration = pulseIn(echoPin, HIGH);    // отримання довжини імпульсу
cm = duration / 58;                   // розрахунок відстані
Serial.print(cm);                     // виведення значення довжини
delay(100);
}
```

---

кінець лістингу 11.1

**Функція `delayMicroseconds()`** зупиняє виконання програми на задану в параметрі кількість мікросекунд. Максимальна пауза, яка коректно відтворюється – 16383. Для зупинки виконання програми більш ніж на кілька тисяч мікросекунд рекомендується використовувати функцію `delay()`.

### **Функція `pulseIn()`**

Функція зчитує довжину сигналу на заданому вході (HIGH або LOW). Наприклад, якщо задано зчитування HIGH функцією `pulseIn()`, функція чекає поки на заданому порту не з'явилося HIGH. Коли HIGH отримано, включається таймер, який буде зупинений коли на вході буде LOW. Функція `pulseIn()` повертає довжину сигналу в мікросекундах. Функція повертає 0, якщо протягом заданого часу (таймауту) не зафіксований сигнал на вході.

Можливі деякі похибки в вимірюванні довгих сигналів. Функція може вимірювати сигнали довжиною від 10 мікросекунд до 3 хвилин.

Синтаксис функції:

*pulseIn(pin, value)*  
*pulseIn(pin, value, timeout)*

де *pin* – номер входу, на якому буде очікуватися сигнал;

*value* – тип очікуваного сигналу (HIGH або LOW);

*timeout* (опціонально) – час очікування сигналу (таймаут) в мікросекундах;

за замовчуванням – одна секунда.

## **Хід роботи**

1. Побудуйте схему підключення до Arduino ультразвукового далекоміра.
2. Запрограмуйте Arduino таким чином, щоб отримувати значення відстані до перешкоди за допомогою ультразвукового далекоміра та виводити результат на комп'ютер.
3. Оформити звіт по роботі. Звіт повинен містити тему та мету роботи, короткі відомості про функції, що розглядаються в роботі, рисунок побудованої схеми підключення та код програми Arduino.